OSKAR: Omnimodal Self-supervised Knowledge Abstraction and Representation

Mohamed Abdelfattah[*,†] **Kaouther Messaoud**[*] **Alexandre Alahi**

École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

{firstname.lastname}@epfl.ch

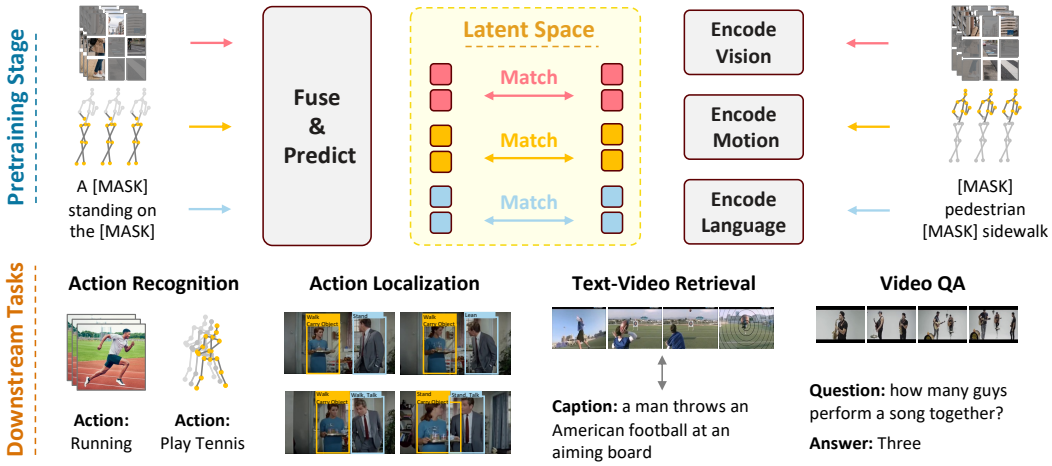


Figure 1: **OSKAR** is a self-supervised multimodal foundation model that learns in the *latent space* using a *fuse-then-predict* strategy. It integrates multiple modalities to capture cross-modal features, matching latent predictions to targets from modality-specific momentum encoders. This preserves uni-modal structure while enabling rich cross-modal learning, and finetuning the unified encoder surpasses specialized models across video, skeleton, and text tasks.

Abstract

We present OSKAR, the first multimodal foundation model based on bootstrapped latent feature prediction. Unlike generative or contrastive methods, it avoids memorizing unnecessary details (*e.g.,* pixels), and does not require negative pairs, large memory banks, or hand-crafted augmentations. We propose a **novel pretraining strategy**: *given masked tokens from multiple modalities, predict a subset of missing tokens per modality, supervised by* ***momentum-updated uni-modal target encoders***. This design efficiently utilizes the model capacity in learning high-level representations while retaining modality-specific information. Further, we propose a **scalable design** which decouples the compute cost from the number of modalities using a fixed representative token budget—in both input and target tokens—and introduces a parameter-efficient cross-attention predictor that grounds each prediction in the full multimodal context. We instantiate OSKAR on video, skeleton, and text modalities. Extensive experimental results show that OSKAR's unified pretrained encoder outperforms models with specialized architectures of similar size in action recognition (rgb, skeleton, frozen, low-shot) and localization, video-text retrieval, and video question answering. Project website: https://multimodal-oskar.github.io

[*]Equal contribution.

[†]Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

1 Introduction

Human perception is inherently multimodal—we naturally integrate visual, motion, and linguistic cues to form coherent understanding from partial observations. In computer vision, multimodal models offer key advantages: (1) they align with human perception by leveraging visual, structural, and semantic signals; (2) they provide architectural flexibility through unified, reusable representations; and (3) they enhance robustness by fusing complementary inputs (*e.g.,* RGB + LiDAR). Existing multimodal methods typically fall into two categories: *generative* and *contrastive*. Generative approaches [6, 60, 7, 36, 42, 26], often based on masked autoencoding [40], focus on low-level reconstruction, which may waste capacity on irrelevant details. Contrastive methods [34, 83, 3, 55, 88] align high-level embeddings but rely on modality-specific priors, handcrafted augmentations, and lack cross-modal predictive reasoning. Therefore, we ask: *Is it possible to move beyond inefficient reconstruction and restrictive contrastive objectives to learn rich cross-modal representations?*

To address these challenges, we explore multiple strategies for effectively routing multimodal information, culminating in **OSKAR** (**O**mnimodal **S**elf-supervised **K**nowledge **A**bstraction and **R**epresentation), comprising three novel contributions:

(1) A new pretext task: *given partial multimodal observations, use cross-modal cues to predict the latent representations of a subset of the missing parts in each modality.* As shown in Fig. 1, OSKAR fuses visible multimodal tokens but infers the missing token representations in each modality separately. This strikes a crucial balance between cross-modal fusion and retention of modality-specific information. Further, by learning in the latent space, the model capacity is efficiently utilized in learning transferrable high-level representations instead of low-level details. Our approach is grounded in the predictive coding theory [65, 30], which posits that the brain learns by predicting internal multimodal representations and minimizing errors, rather than reconstructing raw inputs.

(2) Modality-specific target encoders: Unlike prior works [63, 70, 20, 43] distilled from external teachers, OSKAR *trains from scratch* with momentum-updated target encoders—one per modality. These encoders co-evolve with the model and data, generating stable yet adaptive targets that align closely with the model's internal representation space. This design offers an interesting trade-off: with a fixed momentum update rate, we get shared-weight target encoders, thus providing a flexible *modality-agnostic* encoder in fine-tuning; with customized update rates, we allow each modality to evolve at its own learning pace, providing multiple *modality-specific* encoders with peak performance.

(3) Scalable design: OSKAR scales efficiently thanks to three key design choices. First, it processes a fixed total number of tokens in *both* the input and target, thus dissociating the compute cost from the number of modalities. Importantly, it ensures fair representation of all modalities, regardless of their raw size, through a Dirichlet allocation strategy. Second, it avoids information leaks through non-overlapping masking, *within* and *between* the inputs and targets. Finally, OSKAR introduces a *unified*, modality-agnostic cross-attention predictor that efficiently anchors predictions in shared multimodal context—seamlessly scaling to new modalities with limited growth in model size.

Though OSKAR supports plug-and-play extensibility, we instantiate it on three distinct modalities—video (dense), poses (sparse), and text (symbolic)—forming a challenging testbed for evaluation. Trained *entirely with pseudo-labels* and without manual annotations, OSKAR matches or surpasses specialized models on RGB- and skeleton-based action recognition (86.1% K400, 91.1% NTU120-XSub), spatiotemporal action localization (37.9 mAP AVA), text-video retrieval (50.4 R@1 MSRVTT), and video question answering (49.3% MSRVTT-QA). It also outperforms baselines in low-sample, low-parameter, and low-label settings—highlighting its efficiency and versatility.

2 Related Works

Generative architectures (GAs). Generative self-supervised models corrupt the input and train an encoder together with a lightweight reconstruction head to in-paint the missing content in *input space*. MAE [40] restores masked image patches; VideoMAE [71] extends the idea to spatio-temporal tubes; OmniMAE [35] accepts mixed image–video inputs; Other unified masked models[78, 68] extend it to vision–language and follow-ups [6, 60, 7, 33] broaden the paradigm to additional modalities. While effective at capturing low-level details, these methods optimize pixel-level losses, often diverting model capacity toward reconstructing semantically uninformative elements like texture or lighting.

Joint-embedding architectures (JEAs). JEAs learn by *aligning* representations. Given two or more views of the *same* instance—obtained through data augmentation or drawn from another modality—the model is trained with a contrastive [19, 64], redundancy-reduction [89, 11] or average embedding entropy maximization [15, 4, 38] objective that pulls similar pairs together in feature space while pushing dissimilar ones apart. In cross modal context, CLIP [64] aligns whole-image and sentence embeddings via a pure contrastive loss; More recent models [50, 23, 61, 46] add a momentum teacher that *self-distills* latent knowledge to the student. ImageBind [34] generalises the recipe to six sensory streams. The objective enforces only *global* agreement between paired embeddings. As a result, JEAs excel at zero-shot recognition and retrieval, yet they lack a mechanism for *structured* cross-modal prediction—for instance, nothing in CLIP compels the model to infer a missing video patch from a co-occurring caption.

Joint-embedding predictive architectures (JEPAs). JEPAs [47] blend both worlds by replacing pixel reconstruction with *latent* prediction. A predictor maps visible-context embeddings to the target embeddings of masked regions; the loss is computed in feature space. Hence, the objective is to *learn representations that are predictive of each other*. On images, I-JEPA [5] demonstrates that latent prediction yields strong abstraction with lower compute than MAE [40]. V-JEPA [12] and S-JEPA [1] adapt the idea to video and skeleton data, respectively. iBOT [91] and data2vec [8, 9] also fall under the JEPA framework, performing latent prediction from masked inputs to match teacher representations—at the patch level in iBOT, and in a modality-agnostic manner in data2vec.

OSKAR diverges from these methods in the following: (1) *Cross-modal latent prediction:* Compared to [5, 12, 1], OSKAR *intentionally* uses far fewer per-modality tokens; yet, it complements them with cross-modal cues. Hence, the model is forced to learn from fused cues from **all** modalities to predict the missing latent features in each. (2) *Modality-specific target encoders:* OSKAR employs separate target encoders per modality, balancing intra-modality structure with cross-modal alignment, and allowing a flexible design choice between shared or customized encoders. (3) *Scalable cross-modal masking:* OSKAR masks both the inputs and targets with fixed budgets to keep the compute cost manageable with increasing modalities. Further, it uses Dirichlet sampling to process dense and sparse modalities fairly, while ensuring cross-modal exclusivity to avoid trivial shortcuts.

3 Methodology

Overview. OSKAR operates in two stages: pretraining and fine-tuning. In pretraining (see Fig. 2), OSKAR optimizes a novel "fuse–then–predict" task: given partially masked multimodal tokens, it fuses visible inputs via a cross-modal fusion encoder and predicts modality-specific *latent representations* for masked tokens using shared-weight per-modality predictors. Target representations are generated by momentum-updated encoders, providing stable supervision. Importantly, *learning occurs entirely in the latent space*, emphasizing high-level semantics over low-level reconstruction. In fine-tuning, the target encoders are adapted directly to downstream tasks. All backbones are standard transformers [24], enabling seamless integration into existing transformer pipelines.

3.1 Architecture

Tokenization and Embedding. Following [60, 7], we first tokenize raw inputs (*e.g.,* videos, skeletons, text) using modality-specific encoders [12, 92, 69]; tokenization serves only to accelerate training and is not required for the method itself, producing tokens $\mathbf{T}_m$ for each modality m. These are linearly projected via g_m into a shared embedding space $\mathbf{E}_m \in \mathbb{R}^{N_m \times d}$ of N_m modality tokens, enabling a unified encoder across modalities without task-specific customizations. Each $\mathbf{E}$ is then augmented with learnable positional $e_{(\mathrm{pos})}$, modality $e_{(\mathrm{mod})}$, and auxiliary $e_{(\mathrm{aux})}$ signals. In particular, $e_{(\mathrm{aux})}$ are modality-specific auxiliary signals proposed to resolve ambiguities (*e.g.*, helping the predictor establish pixel-to-keypoint correspondence and disambiguate multiple people in a scene. Without them, the model may associate skeletons with the wrong subject, especially in crowded scenes). We then apply our masking strategy (explained later) to keep only a few non-overlapping patches in the fusion and target encoders, denoted $\mathbf{E}^f \in \mathbb{R}^{N^f \times d}$ and $\mathbf{E}^t \in \mathbb{R}^{N^t \times d}$, where N^f and N^t denote the number of input and target patches, respectively.

Fusion encoder. Given $\mathbf{E}^f$, the fusion encoder's objective is to exploit the complementary cues from *all* modalities to infer the missing information in each. To that end, we feed $\mathbf{E}^f$ to the fusion

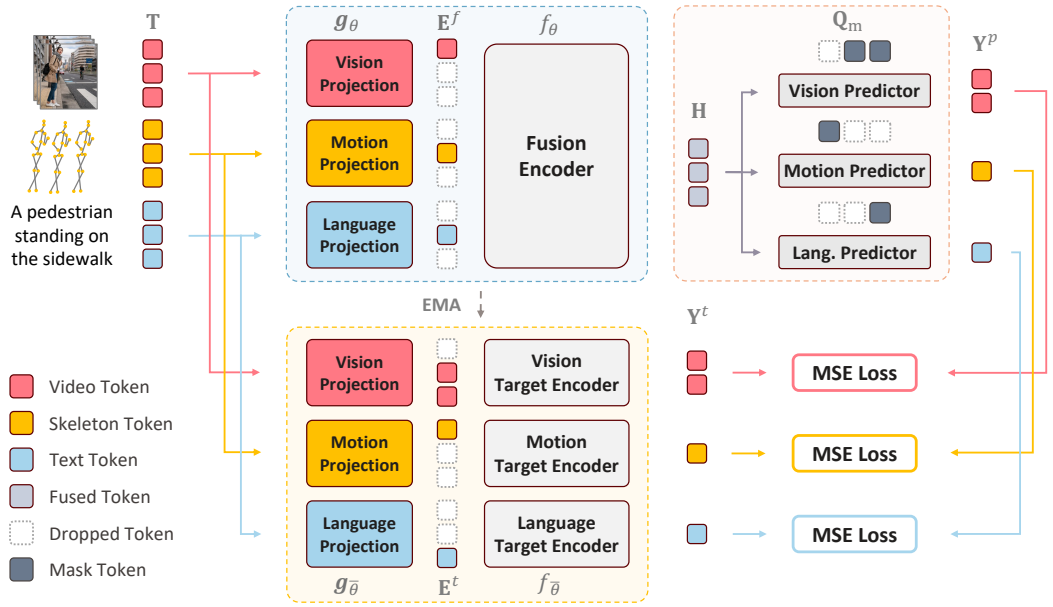


Figure 2: **OSKAR Architecture.** Multimodal tokens (*e.g.,* video, skeleton, text) are projected into a shared space and split into two branches: (1) a fusion encoder processes visible tokens to produce fused representations; (2) modality-specific target encoders generate target embeddings. A predictor estimates masked representations from fused tokens, supervised via MSE loss against targets. Target encoders are updated via EMA of the fusion encoder and are used exclusively in fine-tuning.

transformer f_θ, where all modality tokens interact with each other through the inter-modal Multi-Head Self-Attention (MHSA) mechanism [72], yielding fused representations $\mathbf{H} \in \mathbb{R}^{N^f \times d}$.

Predictor. The role of the predictor is to generate the latent features for a subset of the missing tokens in every modality. These predictions are conditioned on predictor-specific *target-location* queries $\mathbf{Q}^t \in \mathbb{R}^{N^t \times d}$, which are learnable mask tokens $\mathbf{M}^t$, augmented with positional $e^p_{(\text{pos})}$, modality $e^p_{(\text{mod})}$, and auxiliary cues $e^p_{(\text{aux})}$. The predictor processes each set of queries $\mathbf{Q}^t_m$ of modality m through a transformer with alternating self-attention and cross-attention layers. Self-attention allows queries to share information and capture intra-modality structure, while cross-attention integrates information from the fused representations $\mathbf{H}$. At each cross-attention layer, $\mathbf{Q}^t_m$ attend to all tokens in $\mathbf{H}$:

$$\mathbf{Y}^p_m = \text{MHCA}(\mathbf{Q}^t_m, \mathbf{H}, \mathbf{H}), \tag{1}$$

where $\mathbf{Y}^p_m$ are the predicted representations and MHCA is Multi-Head Cross-Attention. Hence, $\mathbf{Y}^p \in \mathbb{R}^{N^t \times d}$ concatenates all per-modality predictions $\mathbf{Y}^p_m$. Importantly, there is a single predictor network with shared weights across modalities. However, it operates like several—one per modality—while retaining the efficiency and regularization benefits of shared weights. This design (i) allows queries to exchange information within a modality (via self-attention), (ii) grounds every prediction in the full multimodal context $\mathbf{H}$ (via cross-attention), and (iii) adapts to new modalities without additional heads, all while remaining decoupled from the fusion encoder for easy transfer to downstream tasks.

Target encoders. Unlike raw inputs (*e.g.,* pose joints)—often subtle, noisy, and isolated—the target encoders provide clean, high-level targets $\mathbf{Y}^t \in \mathbb{R}^{N^t \times d}$, steering the model away from overfitting to spurious details. A key innovation in OSKAR is the use of *modality-specific target encoders*—rather than a single cross-modal target encoder—to balance two objectives: enabling the fusion encoder to learn cross-modal abstractions while preserving each modality's structure and information content. While the fusion encoder and predictor parameters (θ, ϑ) are updated with gradients, each target encoder's parameters $\bar\theta_m$ are updated via an exponential moving average (EMA) of θ:

$$\bar\theta_m \leftarrow \lambda_m \bar\theta_m + (1 - \lambda_m)\theta, \tag{2}$$

where λ_m is a modality-specific momentum coefficient. OSKAR supports two target encoder update strategies: (1) *Shared-weight target encoders* (*i.e.,* same λ for all m): offering a unified target encoder

network with strong downstream multimodal performance, and (2) *Customized target encoders* (*i.e.,* modality-specific λ_m values) offering multiple target encoders with better uni-modal performance, trained with varying update rates that accommodate each modality's learning dynamics. For flexible evaluation, we adopt the first option by default but we study all design choices in ablations sec 5.

3.2 Training

Pretraining: Fuse–then–Predict. Given partially masked multimodal tokens, OSKAR fuses visible inputs with a cross-modal transformer and predicts modality-specific *latent* features for masked tokens via a single shared predictor. Targets are produced by momentum-updated encoders for stable supervision. Learning occurs entirely in latent space, emphasizing high-level semantics over low-level reconstruction.

Pretraining Objective. With both $\mathbf{Y}^p$ and $\mathbf{Y}^t$ now computed, we optimize OSKAR by minimizing a Mean Square Error (MSE) loss between the predicted and target representations:

$$\mathcal{L} = \frac{1}{N^t} \sum_{i \in N^t} \left\| \mathbf{Y}_i^p - \mathbf{Y}_i^t \right\|_2^2 . \tag{3}$$

Crucially, by predicting in feature space, we bypass the need for task/modality-specific losses (*e.g.,* pixel-wise MSE for images or cross-entropy for text). This grants **universal flexibility**: The shared latent objective delivers comparable gradients across modalities, simplifying optimization, reducing negative transfer, and supporting graceful scaling to new modalities without the loss-balancing game.

Cross-Modal Masking Strategy. Multimodality introduces three core challenges: scalability and efficiency with increasing modalities, imbalance because of modality size disparities, and trivial cross-modal shortcuts. OSKAR addresses them all with key design choices: (1) *Fixed token budget:* Inspired by [6], we sample a fixed total budget of N tokens, but for *both* inputs and targets, decoupling the compute cost from the number of input modalities. (2) *Adaptive budgeting:* Instead of naive random sampling—which would let larger modalities (*e.g.,* video) overwhelm smaller ones (*e.g.,* text)—we allocate a fraction r_m of N to each modality m by drawing r_m from a symmetric Dirichlet(α) distribution, ensuring $\sum_m r_m = 1$. Lower α values (*e.g.,* 0.1–0.5) assign all N to a single modality, while higher values (≥ 1) promote more balanced allocations. (3) *Cross-modal exclusivity:* We sample $N_m = r_m \times N$ tokens per modality under a cross-modal spatio-temporal exclusivity constraint, which prevents trivial prediction—if a video patch is visible to the fusion encoder, the corresponding skeleton joint is masked from the target encoder. To reduce redundancy in sequence modalities, we mask contiguous spatio-temporal tubes, encouraging the model to reason over extended, coherent structures rather than isolated tokens. The fusion and target encoder token sets, $\mathbf{X}^f$ and $\mathbf{X}^t$, are sampled independently and are strictly disjoint ($\mathbf{X}^f \cap \mathbf{X}^t = \emptyset$), preventing direct copying and enforcing meaningful prediction of unseen information. This strategy generalizes seamlessly to new modalities while maintaining computational fairness and discouraging shortcuts.

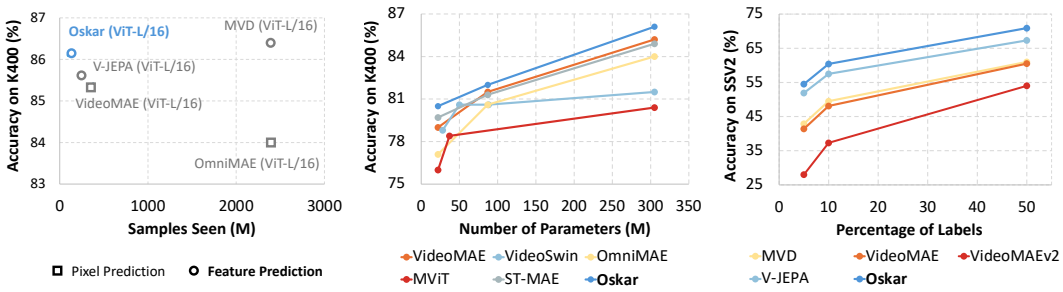
Fine-Tuning. After pretraining, the target encoders are adapted directly to downstream tasks. Because all backbones are standard transformers, fine-tuning integrates seamlessly into existing transformer pipelines.

4 Experimental Results

4.1 Pretraining Details.

Datasets. We train OSKAR **entirely with pseudo-labels** on 10M videos from OpenHumanVid[3] [49] (13.2M videos, 16.7K hours). Using YOLO11 [44], we pseudo-label pose, tracking, and detection, selecting top-3 individuals in crowded videos via visibility, motion, keypoint/bbox confidence, and center proximity. Captions are generated with MiniCPM [87] and CogVLM [41]. Following [60, 7], we speed up processing by pre-tokenizing the videos, skeletons, and text using V-JEPA [12], MotionBERT [92], and WordPiece [69], respectively. Downstream benchmarks include: Kinetics-400 [45] and Something-Something V2 [37] (RGB action recognition), NTU60 [66] and NTU120 [54] (skeleton action recognition), AVA [39] (action localization), MSRVTT [84]/MSVD [16]/VATEX [79] (text-video retrieval), and MSRVTT-QA [84]/MSVD-QA [82]/TGIF-FrameQA [52] (videoQA).

[3]At the submission time of this paper, only 10M videos from OpenHumanVid were publicly released.



(a) Sample Efficiency. (b) Parameter Efficiency. (c) Label Efficiency.

Figure 3: **OSKAR exhibits strong scalability** with (a) fewer samples, (b) less parameters, and (c) less labels per class than comparable methods.

Pre-training. We use standard ViT-S, ViT-B, and ViT-L [24] backbones with learnable positional encodings. All target encoders use a shared EMA update parameter ($\lambda = 0.998$). Models are randomly initialized and trained on 500B tokens (10B warmup) using AdamW [58] ($\beta_1 = 0.9$, $\beta_2 = 0.95$), a base learning rate of 1e-4, cosine decay, batch size 8192, and weight decay 0.05. Transformers use SwiGLU [67] activations and bfloat16 [14] precision. Each model processes $N^s = N^t = 128$ tokens per step, with aggressive masking (<5%, 128 of 2640 tokens visible) via non-overlapping modality masks and Dirichlet-sampled allocation ratios ($\alpha = 0.5$). Pretraining uses 256 GH200 GPUs. Video inputs are 16 frames (stride 2), resized to 224×224; skeletons are temporally aligned and normalized to match video dimensions.

4.2 Main Results

By default, OSKAR is fine-tuned per task using standard inputs (*e.g.,* video only for action recognition).

Action recognition. OSKAR consistently outperforms specialized video-only models across multiple model sizes and datasets without extra cost. On K400 (Tab. 1), OSKAR outperforms V-JEPA [12] and VideoMAE [71] with the same ViT-L backbone, and scales better with limited data (Fig. 3a) and fewer parameters (Fig. 3b). It also beats MViT [27], BEVT [77], and TimeSformer [13] with fewer frames/parameters. For SSv2, gains reach **+3.3%** over VideoMAE (ViT-S) and **+2.5%** over V-JEPA (ViT-L). Beyond video, OSKAR transfers effectively to skeleton-based action recognition: on NTU60 and NTU120 XSub (Tab. 4), OSKAR-B achieves new state-of-the-art results, outperforming MotionBERT [92] by **+0.9%** and **+6.1%**, respectively.

Frozen low-shot action recognition. OSKAR's frozen features deliver strong out-of-the-box performance on SSV2 without fine-tuning, particularly in label-scarce settings (Fig. 3c). With only 5%, 10%, or 50% of the labels, it consistently outperforms other models, achieving a **26.5%** absolute gain over VideoMAEv2 [75] and **2.6%** over V-JEPA [12] at the 5% setting. Unlike V-JEPA's visual-only pretraining, OSKAR *intentionally* reduces the number of visible video tokens but supplements them with motion and language tokens to ground the visual features in semantic and structural information, promoting generalization and yielding consistent gains (**+2.6–3.6%**) across all low-shot settings.

Spatiotemporal action localization. OSKAR transfers effectively to spatiotemporal action localization, consistently outperforming larger video-only models. With just 22M parameters (ViT-S), it matches SlowFast [28] (59M) and surpasses VideoMAE-S by **+5.0** mAP. Scaling up, OSKAR outperforms VideoMAE-B by **+3.5** mAP (ViT-B) and, at the large scale, exceeds V-JEPA and VideoMAE-H by **+1.7** and **+1.4** mAP, respectively—while using less than half the parameters of VideoMAE-H. These gains reflect OSKAR's ability to leverage multimodal pretraining to capture both semantic context and fine-grained motion cues, essential for localizing actions in space and time.

Text-video retrieval. Compared to methods trained with <200M pairs, OSKAR (ViT-L) achieves 50.4 R1 on MSRVTT (**+2.6** over OmniVL [73]); 54.4 R1 on MSVD (**+4.3** over LAVENDER [51]); 54.1 R1 on VATEX (**+3.7** over CLIP4CLIP [59]). Notably, OSKAR performs within a close margin to specialist models trained with 2–3× more data (*e.g.,* only **1.0** below Slide4Video on MSR-VTT) and excels in video-to-text retrieval (**+1.6** to **+5.4** R@1 over CLIP2TV [32]/CenterCLIP [90]/CLIP4Clip [59].

Table 1: **RGB-based action recognition** accuracy (%) on Kinetics-400 [45].

Method	Resolution	GFLOPs	Acc.
Small Models (<80M parameters)			
VideoMAE-S [71]	16×224²	57	79.0
SlowFast+NL [28]	80×224²	234	79.8
MViTv1-B [27]	32×224²	170	80.2
OSKAR-S	16×224²	57	**80.5**
Medium Models (80-150M parameters)			
OmniMAE-B [35]	16×224²	180	80.6
TimeSformer-B [13]	96×224²	2380	80.7
BEVT-B [77]	32×224²	282	81.1
ST-MAE-B [29]	16×224²	180	81.3
VideoMAE-B [71]	16×224²	180	81.5
OSKAR-B	16×224²	180	**82.0**
Large Models (150-700M parameters)			
VideoSwin-L [57]	32×224²	604	83.1
OmniMAE-L [35]	16×224²	597	84.0
VideoMAE-L [71]	16×224²	597	85.2
V-JEPA-L [12]	16×224²	597	85.6
OSKAR-L	16×224²	596	**86.1**

Table 2: **Action detection** mAP on AVA v2.2 [39], all using 16×224² resolution.

Method	PT data	Param (M)	mAP
Small Models (<80M parameters)			
VideoMAE-S [71]	K400	22	22.5
MViTv1-B [27]	K600	36.3	26.1
MViTv2-B [53]	K400	34.5	26.2
SlowFast [28]	K600	59.2	**27.5**
OSKAR-S	OpenHumanVid	**22**	**27.5**
Medium Models (80-150M parameters)			
VideoMAE-B [71]	K400	87	26.7
OSKAR-B	OpenHumanVid	87	**30.2**
Large Models (150-700M parameters)			
VideoMAE-L [71]	K400	305	34.3
ST-MAE-L [29]	K400	304	34.8
VideoMAE-H [71]	K400	633	36.5
V-JEPA-L [12]	VideoMix2M	200	36.2
ST-MAE-L [29]	K700	304	37.3
OSKAR-L	OpenHumanVid	305	**37.9**

Table 3: **Text-to-video retrieval** Recall@1 on MSRVTT [84], MSVD [16], and VATEX [79].

Method	Pairs (M)	MSRVTT	MSVD	VATEX
Methods using large-scale data (>400M samples)				
Cap4Video [81]	400	51.4	51.8	66.6
S4Vid-L[86]	400	51.4	54.9	**67.9**
CLIP-ViP [85]	500	54.2	-	-
IntVideo [80]	646	**55.2**	**58.4**	-
Methods using < 200M samples				
TeachText [21]	-	29.6	25.4	53.2
CLIP4CLIP [59]	100	46.2	-	50.4
Frozen [10]	5	31.0	33.7	-
VIOLET [31]	138	34.5	-	-
SUPPORT [62]	100	30.1	28.4	45.9
LAV. [51]	30	40.7	50.1	-
Singularity [48]	17	42.7	-	-
UMT-B [56]	5	46.3	47.4	-
DRL-B [76]	-	47.6	47.0	44.6
OmniVL [73]	17	47.8	-	-
OSKAR-S	160	45.5	47.5	49.4
OSKAR-B	160	50.1	52.4	50.1
OSKAR-L	**160**	**50.4**	**54.4**	**54.1**

Table 4: **Skeleton-based action recognition** accuracy (%) on NTU60 [66] and NTU120 [54].

Method	Param. (M)	NTU60		NTU120	
		XSub	XView	XSub	XSet
MoBERT [92]	62	93.0	97.2	84.8	86.4
S-JEPA [1]	21	93.1	97.6	90.3	91.3
MaskCLR [2]	62	93.9	97.3	87.4	89.5
PC3D [25]	2	94.1	97.1	86.9	90.3
OSKAR-S	22	93.7	97.3	89.6	89.6
OSKAR-B	86	93.9	97.3	90.9	**92.2**
OSKAR-L	305	**94.3**	**97.8**	**91.1**	92.0

Table 5: **VideoQA** accuracy (%) on MSRVTT-QA [82], MSVD-QA [16], and TGIF [52].

Method	Pairs (M)	MSRVTT	MSVD	TGIF
IntVideo [80]	646	47.1	55.5	72.2
GIT2 [74]	12900	45.6	58.2	74.9
VALOR-L [17]	433	49.2	**60.0**	78.7
COSA [18]	415	49.2	**60.0**	**79.5**
OSKAR-S	160	46.7	56.8	73.2
OSKAR-B	160	48.9	58.3	76.1
OSKAR-L	160	**49.3**	59.7	79.0

Its multimodal pretraining—without external teachers or massive data—outperforms specialized retrieval models of comparable data-parameter scale.

Open-ended Video Question Answering. Without QA-specific architecture modifications, OSKAR demonstrates strong gains on VideoQA benchmarks across two configurations: (1) using OSKAR's visual encoder with a BERT [22] text encoder, and (2) using OSKAR for both video and text encoding. On MSRVTT-QA, OSKAR-L outperforms InternVideo [80] by **+2.2** points and performs on par with VALOR-L [17] (+0.1). On MSVD-QA, it is on par with COSA [18] (–0.3) while surpassing InternVideo by **+4.2**. On TGIF, OSKAR-L performs competitively, coming within 0.5 points of COSA. These results highlight OSKAR's effective general-purpose representations for QA tasks, even with ∼2–4× fewer training pairs.

Qualitative Results. To assess OSKAR's cross-modal predictions, we freeze the pretrained fusion encoder and predictor, and train a lightweight transformer decoder to map features to joint-space coordinates. As shown in Fig. 4, OSKAR accurately reconstructs missing human poses from video and text tokens. Notably, the first column shows that bounding box embeddings effectively guide pose prediction, even in cluttered scenes with multiple people. These results highlight OSKAR's strong multimodal grounding and ability to preserve spatial structure.

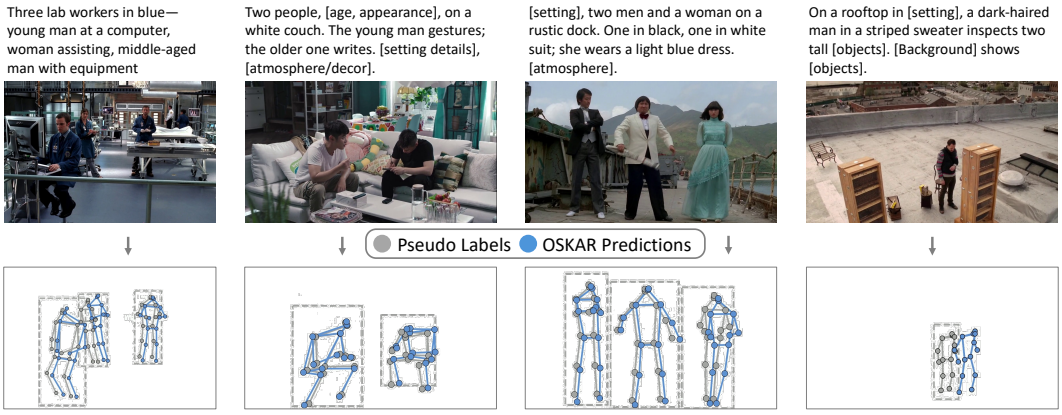
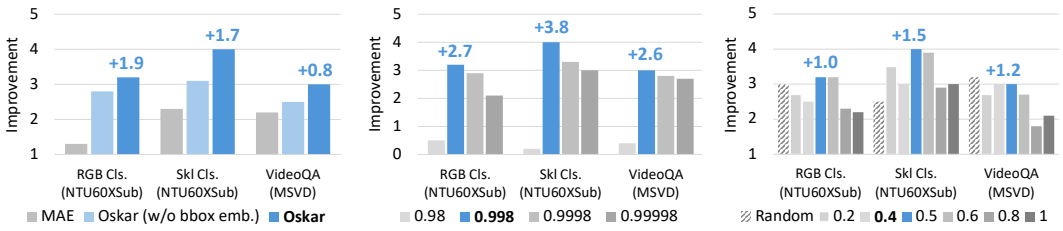


Figure 4: **Visualization of predicted pose features.** OSKAR accurately predicts human poses from video and text, guided by bounding boxes, even in cluttered, multi-person scenes.



(a) MAE and BBox embeddings. (b) EMA update parameter. (c) Concentration Parameter α.

Figure 5: Ablations on (a) MAE and bounding box embeddings, (b) the EMA update parameter, and (c) the α value of the Dirichlet distribution. Blue denotes the default setting of OSKAR. **Blue bold** numbers indicate the difference between our default setting and lowest bar.

5 Ablation Studies

Before large-scale pre-training, we ablate design choices by pre-training ViT-S on 100K OpenHumanVid samples, then fine-tuning on NTU60-XSub [66] for RGB (VidCls) and skeleton (SklCls) action recognition, and MSVD [16] for VidQA, comparing to training from scratch (baseline).

Effect of adding modalities during pre-training. Table 6 shows that combining modalities consistently boosts performance across tasks. Pretraining with video or skeleton alone yields moderate gains (*e.g.,* **+2.6** for VidCls, **+2.9** for SklCls). Adding text with video further improves VidQA results (**+2.1**), underscoring its value for semantic understanding. The best performance (**+3.2/+4.0/+4.3** for VidCls/SklCls/VidQA) comes from using all three modalities, confirming that multimodal grounding of appearance, motion, and language produces more transferable features.

Staged Multimodal Attention Routing. Ablations on OSKAR's attention routing (Table 8) show that the best performance (+3.2 VidCls, +4.0 SklCls, +4.3 VidQA) comes from using this configuration: cross-attention in the fusion encoder to align complementary signals early (*e.g.,* motion and pose), enabling richer representations; intra-modality attention in the target encoder to preserve modality-specific structure, yielding clearer supervision; and the predictor's hybrid setup—first self-attention within modalities, then cross-attention to fused features—balances specialization with contextual grounding (+0.1/-0.3/+0.5 over full self-attention). This staged strategy, integrating early and specializing late, reflects how humans process and combine sensory inputs.

Prediction in the input vs feature space. Figure 5a shows that predicting in feature space outperforms input-space reconstruction (MAE) by +1.9 VidCls, +1.7 SklCls, and +0.8 VidQA. These gains echo the advantages of self-distillation reported in unimodal models [5, 12, 15, 38]. Whereas MAE spends capacity reproducing low-level artefacts such as blur or illumination, OSKAR concentrates on semantic cues—e.g., motion dynamics—and its momentum-updated target encoders further stabilise the supervision signal.

Table 6: **Impact of adding video, skeleton, and text modalities during pre-training.**

Video	Skeleton	Text	VidCls	SklCls	VidQA
	Baseline		88.2	75.7	39.3
✓			+2.6	–	–
✓		✓	+2.2	–	+2.1
	✓		–	+2.9	–
	✓	✓	–	+3.0	–
✓	✓		+2.5	**+4.1**	–
✓	✓	✓	**+3.2**	+4.0	**+4.3**

Table 7: **Shared vs customized target encoders.** Update speeds: + (slow, $\lambda = 0.99998$), ++ (moderate, $\lambda = 0.9998$), +++ (fast, $\lambda = 0.998$).

Video	Skeleton	Text	VidCls	SklCls	VidQA
	Baseline		88.2	75.7	39.3
+	+	+	+3.2	+4.0	+4.3
+++	+	++	+3.2	+3.9	+3.9
+	+++	++	+2.6	+2.9	+3.7
++	+	+++	+3.2	+3.6	+3.8
+	++	+++	+2.8	+3.9	+4.0
+++	**++**	**+**	**+5.0**	**+5.4**	**+5.2**
++	+++	+	+4.5	+4.8	+5.0

Table 8: **Ablation on modality attention routing.** "S": separate; "C": cross-modality.

Fusion	Target	Pred.	VidCls	SklCls	VidQA
	Baseline		88.2	75.7	39.3
S	S	S	+1.1	+3.4	+2.8
S	S	C	+1.3	+2.9	+3.2
S	C	S	+0.8	+2.8	+3.4
C	C	C	+1.2	+2.2	+3.3
C	C	S	+1.0	+2.3	+3.2
C	**S**	**S**	**+3.2**	**+4.0**	**+4.3**

Table 9: **Impact of the number of input and target tokens.**

Input	Target	VidCls	SklCls	VidQA
Baseline		88.2	75.7	39.3
64	64	+1.0	+2.9	+2.7
128	128	+3.2	+4.0	+4.3
256	256	+3.3	+3.3	+3.6
128	256	+3.1	+4.0	+4.2
64	**256**	**+3.4**	**+4.3**	**+4.6**

Shared-weight vs Customized target encoders. The EMA coefficient λ determines how quickly the target encoder tracks the fusion encoder. At $\lambda = 0$, the encoders are identical, causing representation collapse and near-random performance. Too small a λ (*e.g.,* 0.1), i.e., fast updates, destabilizes training, while too large a value (*e.g.,* 0.99998) prevents the target from adapting to new updates. We evaluate two target encoder variants under this trade-off. *(i) Shared-weight target encoders:* A single encoder with a global λ performs best at $\lambda = 0.998$ (+2.7 VidCls, +3.8 SklCls, +2.6 VidQA; Fig.5b). *(ii) Customized target encoders:* Assigning modality-specific λ values—fast for video (0.998), moderate for skeleton (0.9998), and slow for text (0.99998)—yields the best overall results (+5.0 VidCls, +5.4 SklCls, +5.2 VidQA; Table7). This asymmetry aligns with each modality's nature: videos (low variability, high redundancy) benefit from fast updates to focus on motion; skeletons (moderate, structured variation) require balanced updates; and text (high variability, discrete tokens) improves with slower updates to preserve semantic consistency.

Input and target number of tokens. Table 9 shows that more tokens generally improve performance, with the best results at 64 input / 256 target tokens. A 128/128 setting retains 95% of the gains while using less than half the compute, and is thus adopted as default.

Controlling Modality Mix. We analyze the impact of the Dirichlet concentration parameter α on modality token sampling (Fig. 5c). Low α skews sampling toward a single modality, while high α enforces uniformity. Setting $\alpha = 0.5$ yields the best trade-off (+0.2 VidCls, +1.5 SklCls, -0.2 VidQA vs. random). Random sampling favors video due to its token volume, benefiting video tasks, but Dirichlet sampling ensures balanced modality representation, improving SklCls and maintaining competitive video performance. We adopt $\alpha = 0.5$ to promote balanced, modality-aware training.

Bounding box embeddings. Adding bounding box embeddings improves performance across tasks as illustrated in Fig. 5a by providing spatial cues for person-specific predictions. Removing them causes some ambiguity in multi-person settings. These consistent gains highlight the value of simple spatial priors for learning more discriminative representations.

6 Conclusion and limitations

We introduced OSKAR, a novel paradigm for multimodal self-supervised learning that learns semantically rich representations via latent feature prediction. OSKAR introduces a *fuse-then-predict* pretext task, modality-specific momentum encoders for stable supervision, and a scalable masking strategy for balanced and efficient learning. Trained across video, skeleton, and text, OSKAR outperforms specialized models on diverse downstream tasks, while remaining efficient and label-agnostic.

Its modular design supports extensions to new modalities, larger datasets, and adaptive learning dynamics, offering a strong foundation for future multimodal research.

While OSKAR establishes a new state of the art, some limitations offer promising directions for further improvement: (1) *Expanding Modalities:* Although OSKAR currently integrates video, skeleton, and text, adding additional modalities (*e.g.,* audio, depth, IMU) could further enrich the learned representations and unlock new applications. (2) *Scaling Data:* Pretraining on even larger and more diverse datasets would likely enhance the model's generalization and transferability, particularly for complex multimodal reasoning tasks.(3) *EMA Sensitivity:* The performance is sensitive to the choice of the EMA momentum parameter. While this reflects the delicate balance required for stable training, it also highlights an opportunity to develop adaptive or learned momentum strategies to improve robustness. These considerations represent opportunities to build upon OSKAR's significant progress and extend its capabilities even further.

Acknowledgment: This research is funded by the Swiss National Science Foundation (SNSF) through the project grant:10003100. Computational resources were provided as part of the Swiss AI Initiative by a grant from the Swiss National Supercomputing Centre (CSCS) under project ID a03 on Alps.

References

[1] Mohamed Abdelfattah and Alexandre Alahi. S-jepa: A joint embedding predictive architecture for skeletal action recognition. In *European Conference on Computer Vision*, pages 367–384. Springer, 2024.

[2] Mohamed Abdelfattah, Mariam Hassan, and Alexandre Alahi. Maskclr: Attention-guided contrastive learning for robust action representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18678–18687, 2024.

[3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[4] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. Masked siamese networks for label-efficient learning. In *European conference on computer vision*, pages 456–473. Springer, 2022.

[5] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. *arXiv preprint arXiv:2301.08243*, 2023.

[6] Roman Bachmann, David Mizrahi, Andrei Atanov, and Amir Zamir. Multimae: Multi-modal multi-task masked autoencoders. In *European Conference on Computer Vision*, pages 348–367. Springer, 2022.

[7] Roman Bachmann, Oguzhan Fatih Kar, David Mizrahi, Ali Garjani, Mingfei Gao, David Griffiths, Jiaming Hu, Afshin Dehghan, and Amir Zamir. 4m-21: An any-to-any vision model for tens of tasks and modalities. *Advances in Neural Information Processing Systems*, 37:61872–61911, 2024.

[8] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International conference on machine learning*, pages 1298–1312. PMLR, 2022.

[9] Alexei Baevski, Arun Babu, Wei-Ning Hsu, and Michael Auli. Efficient self-supervised learning with contextualized target representations for vision, speech and language. In *International conference on machine learning*, pages 1416–1429. PMLR, 2023.

[10] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1728–1738, 2021.

[11] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

[12] Adrien Bardes, Quentin Garrido, Jean Ponce, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv:2404.08471*, 2024.

[13] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[14] Neil Burgess, Jelena Milanovic, Nigel Stephens, Konstantinos Monachopoulos, and David Mansell. Bfloat16 processing for neural networks. In *2019 IEEE 26th Symposium on Computer Arithmetic (ARITH)*, pages 88–91. IEEE, 2019.

[15] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[16] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 190–200, 2011.

[17] Sihan Chen, Xingjian He, Longteng Guo, Xinxin Zhu, Weining Wang, Jinhui Tang, and Jing Liu. Valor: Vision-audio-language omni-perception pretraining model and dataset. *arXiv preprint arXiv:2304.08345*, 2023.

[18] Sihan Chen, Xingjian He, Handong Li, Xiaojie Jin, Jiashi Feng, and Jing Liu. Cosa: Concatenated sample pretrained vision-language foundation model. *arXiv preprint arXiv:2306.09085*, 2023.

[19] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PmLR, 2020.

[20] Yanbei Chen, Yongqin Xian, A Koepke, Ying Shan, and Zeynep Akata. Distilling audio-visual knowledge by compositional contrastive learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7016–7025, 2021.

[21] Ioana Croitoru, Simion-Vlad Bogolin, Marius Leordeanu, Hailin Jin, Andrew Zisserman, Samuel Albanie, and Yang Liu. Teachtext: Crossmodal generalized distillation for text-video retrieval. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11583–11593, 2021.

[22] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[23] Xiaoyi Dong, Jianmin Bao, Yinglin Zheng, Ting Zhang, Dongdong Chen, Hao Yang, Ming Zeng, Weiming Zhang, Lu Yuan, Dong Chen, et al. Maskclip: Masked self-distillation advances contrastive language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10995–11005, 2023.

[24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[25] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2969–2978, 2022.

[26] David Fan, Jue Wang, Shuai Liao, Zhikang Zhang, Vimal Bhat, and Xinyu Li. Text-guided video masked autoencoder. In *European Conference on Computer Vision*, pages 282–298. Springer, 2024.

[27] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6824–6835, 2021.

[28] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019.

[29] Christoph Feichtenhofer, Yanghao Li, Kaiming He, et al. Masked autoencoders as spatiotemporal learners. *Advances in neural information processing systems*, 35:35946–35958, 2022.

[30] Karl Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2): 127–138, 2010.

[31] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. Violet: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*, 2021.

[32] Zijian Gao, Jingyu Liu, Weiqi Sun, Sheng Chen, Dedan Chang, and Lili Zhao. Clip2tv: Align, match and distill for video-text retrieval. *arXiv preprint arXiv:2111.05610*, 2021.

[33] Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens Van Der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16102–16112, 2022.

[34] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15180–15190, 2023.

[35] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Omnimae: Single model masked pretraining on images and videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10406–10417, 2023.

[36] Yuan Gong, Andrew Rouditchenko, Alexander H Liu, David Harwath, Leonid Karlinsky, Hilde Kuehne, and James Glass. Contrastive audio-visual masked autoencoder. *arXiv preprint arXiv:2210.07839*, 2022.

[37] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850, 2017.

[38] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[39] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6047–6056, 2018.

[40] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[41] Wenyi Hong, Weihan Wang, Ming Ding, Wenmeng Yu, Qingsong Lv, Yan Wang, Yean Cheng, Shiyu Huang, Junhui Ji, Zhao Xue, et al. Cogvlm2: Visual language models for image and video understanding. *arXiv preprint arXiv:2408.16500*, 2024.

[42] Po-Yao Huang, Vasu Sharma, Hu Xu, Chaitanya Ryali, Yanghao Li, Shang-Wen Li, Gargi Ghosh, Jitendra Malik, Christoph Feichtenhofer, et al. Mavil: Masked audio-video learners. *Advances in Neural Information Processing Systems*, 36:20371–20393, 2023.

[43] Fushuo Huo, Wenchao Xu, Jingcai Guo, Haozhao Wang, and Song Guo. C2kd: Bridging the modality gap for cross-modal knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16006–16015, 2024.

[44] Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024.

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

[46] Sanghwan Kim, Rui Xiao, Mariana-Iuliana Georgescu, Stephan Alaniz, and Zeynep Akata. Cosmos: Cross-modality self-distillation for vision language pre-training. *CVPR*, 2025.

[47] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[48] Jie Lei, Tamara L Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. *arXiv preprint arXiv:2206.03428*, 2022.

[49] Hui Li, Mingwang Xu, Yun Zhan, Shan Mu, Jiaye Li, Kaihui Cheng, Yuxuan Chen, Tan Chen, Mao Ye, Jingdong Wang, et al. Openhumanvid: A large-scale high-quality dataset for enhancing human-centric video generation. *Proceedings of the IEEE international conference on computer vision*, 2025.

[50] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021.

[51] Linjie Li, Zhe Gan, Kevin Lin, Chung-Ching Lin, Zicheng Liu, Ce Liu, and Lijuan Wang. Lavender: Unifying video-language understanding as masked language modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23119–23129, 2023.

[52] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. Tgif: A new dataset and benchmark on animated gif description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4641–4650, 2016.

[53] Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4804–4814, 2022.

[54] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2684–2701, 2019.

[55] Yunze Liu, Qingnan Fan, Shanghang Zhang, Hao Dong, Thomas Funkhouser, and Li Yi. Contrastive multimodal fusion with tupleinfonce. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 754–763, 2021.

[56] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3042–3051, 2022.

[57] Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3202–3211, 2022.

[58] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[59] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021.

[60] David Mizrahi, Roman Bachmann, Oguzhan Kar, Teresa Yeo, Mingfei Gao, Afshin Dehghan, and Amir Zamir. 4m: Massively multimodal masked modeling. *Advances in Neural Information Processing Systems*, 36:58363–58408, 2023.

[61] Muhammad Ferjad Naeem, Yongqin Xian, Xiaohua Zhai, Lukas Hoyer, Luc Van Gool, and Federico Tombari. Silc: Improving vision language pretraining with self-distillation. In *Computer Vision – ECCV 2024*, pages 38–55, Cham, 2025. Springer Nature Switzerland.

[62] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander Hauptmann, Joao Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. *arXiv preprint arXiv:2010.02824*, 2020.

[63] Gorjan Radevski, Dusan Grujicic, Matthew Blaschko, Marie-Francine Moens, and Tinne Tuytelaars. Multimodal distillation for egocentric action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5213–5224, 2023.

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

[65] Rajesh PN Rao and Dana H Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87, 1999.

[66] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[67] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

[68] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15638–15650, 2022.

[69] Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. Fast wordpiece tokenization, 2021.

[70] Zineng Tang, Jaemin Cho, Hao Tan, and Mohit Bansal. Vidlankd: Improving language understanding via video-distilled knowledge transfer. *Advances in Neural Information Processing Systems*, 34:24468–24481, 2021.

[71] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. In *Advances in Neural Information Processing Systems*, pages 10078–10093. Curran Associates, Inc., 2022.

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

[73] Junke Wang, Dongdong Chen, Zuxuan Wu, Chong Luo, Luowei Zhou, Yucheng Zhao, Yujia Xie, Ce Liu, Yu-Gang Jiang, and Lu Yuan. Omnivl: One foundation model for image-language and video-language tasks. *Advances in neural information processing systems*, 35:5696–5710, 2022.

[74] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022.

[75] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14549–14560, 2023.

[76] Qiang Wang, Yanhao Zhang, Yun Zheng, Pan Pan, and Xian-Sheng Hua. Disentangled representation learning for text-video retrieval. *arXiv preprint arXiv:2203.07111*, 2022.

[77] Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. Bevt: Bert pretraining of video transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14733–14743, 2022.

[78] 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 vision and vision-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19175–19186, 2023.

[79] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4581–4591, 2019.

[80] Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

[81] Wenhao Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. Cap4video: What can auxiliary captions do for text-video retrieval? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10704–10713, 2023.

[82] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653, 2017.

[83] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021.

[84] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016.

[85] Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. Clip-vip: Adapting pre-trained image-text model to video-language representation alignment. *arXiv preprint arXiv:2209.06430*, 2022.

[86] Huanjin Yao, Wenhao Wu, and Zhiheng Li. Side4video: Spatial-temporal side network for memory-efficient image-to-video transfer learning. *arXiv preprint arXiv:2311.15769*, 2023.

[87] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

[88] Xin Yuan, Zhe Lin, Jason Kuen, Jianming Zhang, Yilin Wang, Michael Maire, Ajinkya Kale, and Baldo Faieta. Multimodal contrastive training for visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6995–7004, 2021.

[89] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International conference on machine learning*, pages 12310–12320. PMLR, 2021.

[90] Shuai Zhao, Linchao Zhu, Xiaohan Wang, and Yi Yang. Centerclip: Token clustering for efficient text-video retrieval. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, July 11–15, 2022, Madrid, Spain*, 2022.

[91] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

[92] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction (Sec. 1) clearly state OSKAR's contributions. The impacts of these contributions are carefully studied in the experiments (Sec 4) and ablations (Sec. 5) sections.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Sec 6, we provide limitations of our work. We also include a broader impact statement in the Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Detailed pretraining and evaluation settings are provided in Sec 4 and the Appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets used in our experiments are all publicly available. An anonymized version of our code is available as part of the supplementary materials. Codes will be open-sourced to the community upon publication.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Detailed experimental settings are available in section 4 and the Appendix. We also ablate on all introduced hyper-parameters, detailed in section 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We fix a random seed for all statistical values in our experiments, and we follow the best practices from domain knowledge and previous works in our experiments. Due to the high cost of running experiments, we are unable to run each experiment multiple times. Hence, no error bars are included in our plots.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The information on the required compute cost is availabe in section 4 as well as the supplementary materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper abides by the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the societal impacts of our paper in the Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We described the potential impacts of our paper in the Appendix. We will follow the standard safeguards before releasing any models or data.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We carefully cited all the previous models and datasets we used in our work, which are all accessible for research purposes.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: All our assets are well-documented.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: Our work does not invlove any human subjects or crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our work does not invlove any human subjects or crowdsourcing.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.