Learning Video Representation from Self-supervision

Brian Chen

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
under the Executive Committee
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2023

Abstract

Learning Video Representation from Self-supervision

Brian Chen

This thesis investigates the problem of learning video representations for video understanding. Previous works have explored the use of data-driven deep learning approaches, which have been shown to be effective in learning useful video representations. However, obtaining large amounts of labeled data can be costly and time-consuming. We investigate self-supervised approach as for multimodal video data to overcome this challenge. Video data typically contains multiple modalities, such as visual, audio, transcribed speech, and textual captions, which can serve as pseudo-labels for representation learning without needing manual labeling. By utilizing these modalities, we can train deep representations over large-scale video data consisting of millions of video clips collected from the internet. We demonstrate the scalability benefits of multimodal self-supervision by achieving new state-of-the-art performance in various domains, including video action recognition, text-to-video retrieval, and text-to-video grounding.

We also examine the limitations of these approaches, which often rely on the association assumption involving multiple modalities of data used in self-supervision. For example, the text transcript is often assumed to be about the video content, and two segments of the same video share similar semantics. To overcome this problem, we propose new methods for learning video representations with more intelligent sampling strategies to capture samples that share high-level semantics or consistent concepts. The proposed methods include a clustering component to address

false negative pairs in multimodal paired contrastive learning, a novel sampling strategy for finding visually groundable video-text pairs, an investigation of object tracking supervision for temporal association, and a new multimodal task for demonstrating the effectiveness of the proposed model. We aim to develop more robust and generalizable video representations for real-world applications, such as human-to-robot interaction and event extraction from large-scale news sources.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I am deeply grateful for the guidance and support provided by Professor Shih-Fu Chang throughout the completion of this thesis. His exceptional leadership of the Digital Video and Multimedia lab (DVMM) and his ongoing contributions to science and education have been an inspiration to me. The achievements reported in this thesis are direct results of the collaborative efforts of several renowned researchers and dear colleagues, particularly Professor Hilde, for her contributions in Chapters 2 and 3. I would also like to extend my thanks to Bo Wu, Hanwang Zhang, and many others who provided valuable support throughout my journey. Additionally, I am grateful for the support provided by organizations such as DARPA, IBM, and Salesforce. Lastly, I would like to express my deepest gratitude to my family, especially my wife for her unwavering support, and my son for his energy and inspiration.

Chapter 1: Introduction

1.1 Background

Video understanding is a challenging area of research in computer vision that aims to analyze and reason about the content in videos. While images contain static frames, videos offer a wealth of information, including motions, dynamic scenes, and various viewpoints. However, processing videos requires significant computational and storage resources due to their large volumes high-dimensional data. Representations for videos are crucial for these understanding tasks, as they allow us to recognize high-level concepts in the video, such as different actions performed by various objects and temporal relations between events.

Recent advances in deep learning, specifically convolutional neural networks, have achieved great success in learning the encoding of videos, formally known as deep representations. These models extract useful properties of the input data that represent the general nature of the video. However, most state-of-the-art models for video representation rely on large, carefully labeled datasets for training. Additionally, video annotation is expensive and time-consuming, as it is difficult to annotate temporal boundaries and define label classes of events.

To overcome these limitations, self-supervised learning has emerged as a promising direction for video understanding [1, 2, 3]. Rather than relying on human-annotated labels for specific tasks, self-supervised learning aims to learn representations through pretext tasks, such as predicting parts of the data that have been withheld or discriminating pairs of data associated with each other from data that are randomly paired. The hypothesis is that by accomplishing these tasks, the model develops a certain visual understanding ability that results in meaningful representations, which in turn can be used for various downstream tasks. This approach allows for larger-scale training, accessibility to different domains, and avoids the ambiguity of labels [4].

1.2 Motivation for multimodal video self-supervised learning

Recently, self-supervised learning in images has achieved great success and has been shown to outperform supervised pre-training in various downstream tasks, such as image classification and object detection [5, 6]. However, learning from images has its limitations for several reasons. First, images are static, which means they cannot capture causal and temporal information. Additionally, images serve as a unimodal representation where the data itself may be corrupted or noisy in the real world. Furthermore, image self-supervised learning often relies on some forms of human-defined data augmentations such as cropping, rotation, and blurring, which means its supervision does not exist naturally.

In contrast, videos capture temporal dynamics with naturally existing multimodal supervision, such as optical flow, audio, and ASR text. Videos on the internet, such as YouTube, provide audio and ASR captions which can be used to define self-supervised tasks involving content from multiple modalities. Learning from such multimodal supervision has become a promising learning pipeline to train neural networks from scratch without human annotation. Furthermore, learning from multimodal data has the benefit of creating a common space across modalities, some of which such as speech and text offer an opportunity to capture a higher level of semantic concepts. With the recent success of a multi-modal representation, CLIP [7], it is possible to perform recognition tasks for free, by matching the visual content and possible labels in the shared common space.

In this thesis, we focus on developing models for learning such multimodal common spaces, where the model is free from fine-tuning of human-defined labels. We aim to leverage the self-supervision across multiple modalities available in videos to improve the robustness and generalizability of the models in real-world scenarios.

1.3 Overview of learning from multimodal self-supervision

This thesis is primarily focused on the task of learning representations through multi-modal self-supervision. In this section, we will discuss the state of the art in this field and how our work

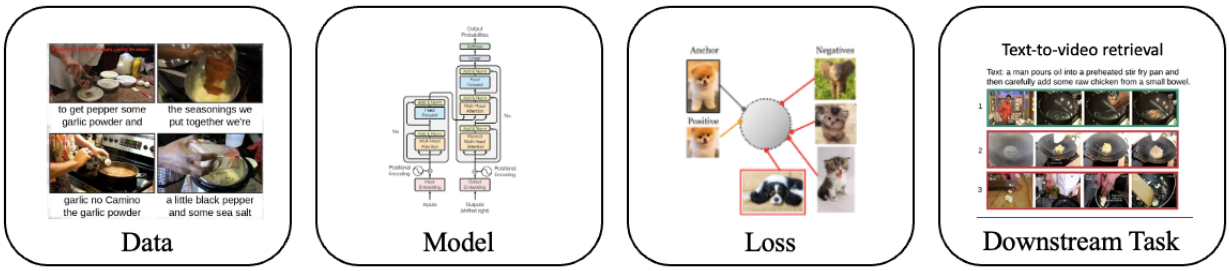


Figure 1.1: Key Ideas for Improvement in Multi-Modal Self-Supervision.

relates to it. Self-supervised learning is a powerful approach for learning representations from data. However, as shown in Figure 3.1 there are several key components that need to be considered, including the data, model, learning objective, and downstream evaluation.

Data. There are two important aspects we need to consider about data for self-supervised learning - data transformations (augmentation) and learning from different domains. Data transformations are key to image-based self-supervised learning [6] where different data augmentations are applied to images such as random cropping, gaussian blur, rotations, and color jittering. The model will treat different transformations of the same datapoint as an identical instance. The objective of the self-supervising task is to map transformed data to the original data in the learned representation space. In the video, different speeds [8] and shuffling data order [9] may serve as the additional transformation of data. These data transformations serve as priors on the types of invariances and distinctiveness to encode in the learned representation. This thesis explores the rich space of data transformations that are possible for video data and uses it for multi-modal self-supervised representation learning. One key data transformation we explore is the viewpoint variation and occlusion of an object captured from video input (chapter 4). We also explore how other modalities, such as automatic speech recognition outputs (chapter 2,3,5), audio (chapter 2,3), and corresponding articles (chapter 5), can be used as powerful data augmentation for representation learning.

We also explore the use of different data domains for representation learning. Previous works primarily focused on learning video representations from video datasets with a fixed set of pre-defined label classes such as Kinetics-400 [10] and Audioset [11]. In addition, we investigate the potential of large-scale multimodal datasets such as HowTo100M [12] and WebVid-2M [13] for representation

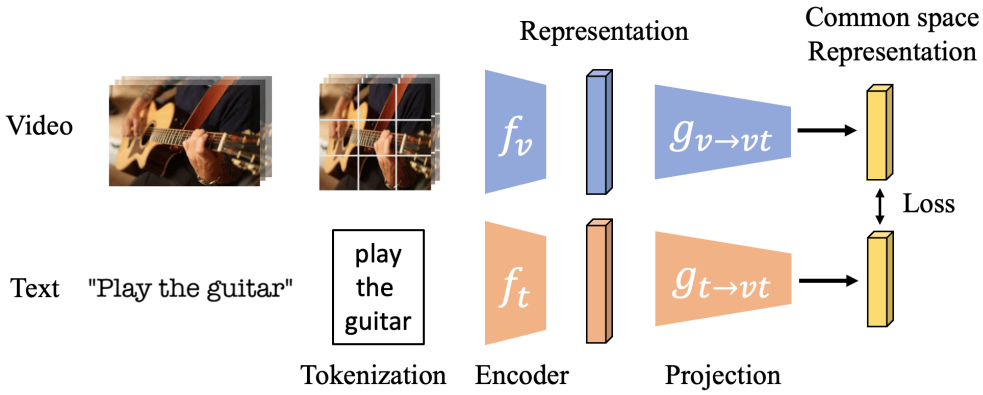


Figure 1.2: Multimodal self-supervised learning pipeline.

learning. In particular, the HowTo100M dataset [12] collects videos with corresponding audios and generates the text transcript by YouTube ASR API back in 2019 automatically, where the ASR tool had a word error rate (WER) of 28% [14]. In this thesis, the text transcript used in the experiments for (chapter 2 and 3) are from the ASR provided in the HowTo100M dataset [12]. In addition, we sampled video segments with ASR output, and the start/end time was bounded by ASR, resulting in a 3-10 seconds video following [2, 12]. Through the use of multi-modal self-supervision, we are able to bypass the need for collecting semantic labels for these datasets and instead utilize the free supervision present in the audio and text modalities. In particular, we explore the News domain where the text is less likely to be related to the video and show that with a sufficiently large data scale, we can perform representation learning on an open-domain dataset to train video representation achieving strong performance for event and argument role labeling (chapter 5)). In this dataset, we applied the YotTube ASR in 2021, where the WER is 20.6% [15].

Models. The model maps the input data into a data representation. In the case of multimodal self-supervision, there are usually tokenizers that are used to process the video into patches and sentence to words, followed by encoders for each modality, as shown in Figure 1.2. Traditionally, video representations were extracted in frame-based encoding using 2D convolutional neural networks [16] , aggregated over time by average pooling or NetVLAD [17] to acquire the frame-based features over time. Recently, video representations were extracted by 3D convolutional neural networks [18] to encode short-term temporal dynamics. Both architectures aim to encode the video's short- and

long-term temporal information. In this thesis, we applied widely used video encoder backbone S3D [19], R(2+1)D [20], C3D [21] for multimodal self-supervied learning. For the text model, word2vec [22] and BERT models [23] are widely used to encode both word and sentence level representations. For the audio model, raw audio [3] or log-mel spectrograms [24] are used as input to ResNet [2] and DaveNet [25].

Transformers [26] have achieved great success in the natural language processing (NLP) field by its multi-head attention and encoder-decoder architecture to compute contextualized representations from a sequence. In this thesis, we demonstrate that multimodal learning can benefit from using a joint multimodal transformer (chapter 5) for computing multimodal contextual representations for video and text rather than disjointed encoders from each modality. We showed the transformer architecture is useful for encoding temporal information on top of extracted visual representations for better multimodal representation in event extraction (chapter 2) and common space learning in zero-shot tasks (chapter 2). The learned representation from the encoder can be represented by a single vector [1]. In Transformer model encoders, the representation mostly maintain consistent dimensions between the spatio-temporal representation for video and the representation for text [27].

Followed by the encoder for each modality, we learn networks that project the representation from each modality to a common space to let the representations to be directly comparable as shown in Fig 1.2. This network was usually implemented as one or two linear layers for projection [28, 1, 3].

Loss Function. The choice of the loss function is extremely important for learning a strong video representation. The loss function drives the learning process and is a key component of the self-supervised learning pipeline. Traditionally, the binary cross-entropy or triplet (max-margin) objectives [29] were used in the multimodal self-supervised learning setup. These losses encouraged the representation to discriminate visual and aural data pair from the same video segment from other pairs. Recently, noise contrastive training [30] has gained popularity to learn self-supervised image representations [6, 5], where the model learns the discriminability by pulling the same instance with different data augmentation to be close and pushing different instances away. In this thesis, we

explore how noise contrastive training can be adapted in the multimodal setting both for video-text (chapter 5) and video-audio (chapter 2 and 3) representation learning. We show that this loss is crucial for learning strong video representations across a variety of domains and tasks (chapter 5). Cluster based representation learning, on the other hand, is to learn an embedding on a unit-sphere that optimizes clusters while maximizing the selected objective function. In [31], a simple alternating procedure was developed to train a model by switching between training a model on pseudo labels derived from clustering results and assigning pseudo labels based on k-means clustering. Incorporating transformations like those mentioned earlier allows the network to become invariant to such transformations and is usually implemented by extracting and amplifying the initial feature extractor (a randomly initialized neural network). In this thesis, we build on this work by extending to multimodalities, where we learn to predict multimodal cluster centroids for better common space learning (chapter 2).

Performance evaluation. Generally, there are two differente ways of evaluating a trained model [32]: by finetuning the weights or keeping the weights of the model frozen. For the former, the pretrained model is further optimized on some human-labeled datasets such as UCF-101 [33], HMDB-51 [34] and Kinetics-400 [10] for action classification where the accuracy was reported. For the latter, the trained model is frozen, and typically only a weak model, such as a linear classifier, is added and trained on some labeled dataset. For video datasets, evaluation is conducted on the UCF-101 [33] and HMDB-51 [34] and includes also nearest neighbor based class retrieval. It is also possible to evaluate without further training a linear layer: the individual representations are saved and k-nearest neighbor (kNN) retrieval [35] is performed. In the multimodal scenario where video and text are available, both the encoder and projection layer can be frozen and perform text-to-video retrieval in datasets such as MSR-VTT [36], VATEX [37], YouCook [38], and typically performance at various neighborhood levels is reported. Note that during evaluation, instead of feeding the ASR transcript to the text branch as training, the human-annotated text description where provided. In the Video M2E2 dataset we have developed (see chapter 5), the event extraction and argument role labeling can be done by feature similarity computation across modalities (chapter 5). Also, we

can perform grounding from text to spatio-temporal regions using the frozen models without any finetuning (chapter 3).

1.4 Technical challenges of learning from mulitmodal self-supervision.

One of the key assumptions of learning from multimodal self-supervision is that the association across modalities is correct. For example, it is assumed that the ASR transcript is about the video content in the same time frame. This assumption leads to the widely used contrastive loss approach, where associated pairs are pulled closer and misaligned pairs are pushed apart, regardless of their semantic meanings. However, this assumption is not always true, as information in ASR may not be relevant to what's shown in video or refer to video content at different times. [1].

In this thesis, we explore various directions to reduce the noise that may have been introduced by this association assumption, with an aim to provide better supervision for self-supervised learning. In chapter 2, we design a clustering component to alleviate the false negative pairs introduced by multimodal paired contrastive learning. In chapter 3, we propose a novel sampling strategy to find visually groundable video-text pairs for better self-supervised grounding supervision. In video-only self-supervised learning, previous works have explored temporal association across time, where it is assumed that visual instances across time are the same. In chapter 4, we investigate object tracking supervision to ensure that the temporal association focuses on the same visual concept. In the last chapter 5, we propose a new multimodal task to demonstrate our model, which learns the multimodal common space from self-supervision, resulting in better video event extraction and argument role labeling performance compared to traditional contrastive loss methods with that rely on the noisy association assumption.

1.5 Thesis Outline and Contributions

In this thesis, we present several novel methods for improving multimodal self-supervised learning in the context of video understanding. Our main contributions are summarized as follows:

Chapter 2: We propose a multimodal self-supervised learning approach that utilizes video, text,

and audio as input modalities. We demonstrate the scalability of this approach by training on a large-scale video dataset, the HowTo100M, consisting of over 120 million video clips. We also aim to alleviate the problem of false negatives in noise contrastive learning by adding a clustering loss to the learning objective. By forcing the model to predict the cluster center of multimodal representation across video, audio, and text embeddings, we guide the model to capture high-level semantics shared among different modalities.

Chapter 3: We demonstrate the benefits of using representations with finer granularity (region level) in multimodal self-supervised learning for spatio-temporal grounding. By incorporating both global and local representations learned from multimodal data, we show that these two types of representations from multiple modalities are complementary with each other, guiding each other for better representation. We also propose a new sampling strategy where we select visually groundable pairs for computing the contrastive loss for higher quality supervision. Additionally, we introduce a new dataset, GroundingYoutube, a multimodal instructional video dataset with both spatial and temporal annotations for video grounding evaluation. Finally, we show that our model can generalize to datasets in other domains besides YouTube instructional videos, such as VGG Sound, without learning from human-annotated data.

Chapter 4: We propose PreViTS, a contrastive pre-training strategy that utilizes unsupervised video object tracking for learning video representations. We show that such tracking supervision allows us to focus on the visual semantics of the same object across time, which sets state-of-the-art performance on various video action recognition datasets. Our representation learned various video invariances such as occlusion and viewpoint. We also demonstrate that our representation focuses more on the foreground actions in the video, less biased towards background information.

Chapter 5: We propose a novel task of Video Multimodal Event Extraction, which aims to automatically extract events and argument roles from video article pairs utilizing video self-supervised learning with multimodal common space. We demonstrate our multimodal self-supervised model learned from video and ASR text can perform multimodal event coreference resolution. Our model is able to select video segments and sentence pairs that refer to the same event. In addition, our

transformer model can perform multimodal event extraction/argument role labeling by jointly decoding the event/argument from both modalities. We show that a model trained with self-supervision can achieve comparable results to a supervised model in extracting events by multimodal common space. Furthermore, we demonstrate that the learned common space can adapt to new domains without human annotation. This approach has the potential to greatly improve the scalability and accessibility of video understanding tasks, as it does not rely on human-annotated data.

Chapter 2: Learning Video Representation from Video, Audio, and Text

2.1 Introduction

To robustly learn visual events and concepts, humans seldom rely on visual inputs alone. Instead, a rich multimodal environment is utilized for understanding by combining multiple sensory signals along with various language representations. Many recent techniques have attempted to mimic this paradigm to train efficient computer vision models, especially those that learn from videos where multiple modalities are naturally present [2, 39, 40].

Learning on multimodal video data has both benefits and challenges. It is beneficial that each video instance has information available in multiple modalities. Textual information corresponding to the spoken narrations in the video, for example, provides a valuable language modality in addition to the visual and audio modalities [41, 25, 42]. In this work, we focus on the problem of learning a joint embedding space across multiple modalities. Given that the features from different modalities are often not comparable, the goal is to learn the projections into a common space where features from different domains but with similar content are close to each other to allow for a direct retrieval across modalities. However, creating an effective joint multimodal embedding space is not easy. First, each of those modalities is different, *i.e.*, with respect to its source, how it is sampled and processed, and its resulting feature representation. Additionally, in real-world data, the supervision available to learn these projections from each of the modalities is unfortunately weak, as *e.g.*, audio sequences can be misaligned to their visual representations and corresponding narration might or might not be present in the same time interval [39, 1].

To deal with multimodal data of this nature, several recent approaches use a contrastive loss [43, 44] to learn e.g. feature representations in a joint embedding space. The goal is to bring samples drawn from the same temporal instance closer to each other while keeping samples from different

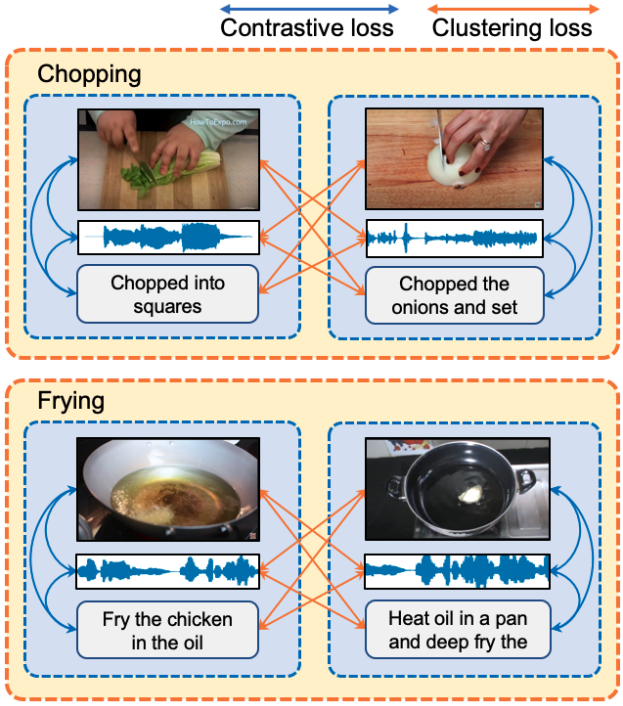


Figure 2.1: The Multimodal Clustering Network (MCN) combines a contrastive loss that learns feature representations to be close across different modalities such as video, audio, and text (blue box), with a clustering loss that draws instances that are semantically related together, e.g., scenes depicting the same semantic concept (e.g., chopping or frying) from different videos or different clips. (yellow box).

times apart. Recent works [2, 1] show that such training is useful for pretraining models on large-scale data without additional supervision and that the resulting models achieve competitive performance on several tasks, *e.g.*, in action classification when fine-tuned on various datasets. One problem arising from the contrastive loss is that this criterion does not consider the samples' semantic structure and similarity at different times: two samples are treated as a negative pair as long as they occur at different times regardless of their semantic similarity. This can have a considerable adverse impact on the learned representation. In a different formulation for learning representations, instead of comparing individual instances, clusters of instances are first created using a certain clustering algorithm [39, 45, 46, 47]. This approach encourages samples semantically similar to each other (namely, samples in the same cluster) to be close in the embedding space. However, if we cluster features from multi-modalities, those clusters would likely emerge only within the modalities separately, clustering audio instances with audio instances, visuals to visuals . Therefore,

a mechanism that pulls the instances from different modalities together is crucial to cluster features from different modalities in a joint space. This leads to our proposed method that treats these two approaches as reciprocal information.

We present a multimodal learning framework that learns joint representations by training cross-modal projection heads from the visual, audio, and language modalities and accounts for the semantic similarity of embedding using a large corpus of naturally narrated videos. The proposed *Multimodal Clustering Network* (MCN) adopts a novel architecture to combine promising ideas from both representation learning paradigms described earlier: learning via the contrastive loss at the instance level and the semantic consistency at the cluster level. As another novel feature of our approach, we explore joint clusters using multimodal representations instead of clusters using separate modalities. The result features allow us to do retrieval across different modalities in linear time. Figure 3.1 provides a high-level overview of our approach.

To evaluate our proposed method, we address the challenging problem of zero-shot learning in two contexts: multimodal video retrieval and multimodal temporal action localization. We train our system on the HowTo100M dataset [12] and evaluate its retrieval capabilities on the YouCook2 [48] and MSR-VTT [36] dataset and its temporal action localization on the task of action detection on the CrossTask [49] dataset and on the task of temporal action segmentation on the Mining YouTube [50] dataset. Using only features from pretrained backbones, MCN significantly outperforms the best text-to-video retrieval baseline over absolute 3% in recall and outperforms the temporal action localization baseline over 3.1% in recall, both in zero-shot settings.

The contributions of this chapter are threefold: (i) We propose a novel method by combining the benefits of contrastive loss and clustering loss for multimodal joint space learning. Unlike prior works that create clusters using separate modalities, our method shows the important benefits of using multimodal joint clusters. (ii) We show that the proposed model can learn across three modalities (video, audio, text) in a joint space. (iii) We demonstrate significant performance gains on multiple downstream tasks in the zero-shot setting. These results show that the learned common space representations can improve state-of-the-art results without any additional training on the

target datasets.

2.2 Related Work

Learning from Multimodal Data. Instead of collecting new annotated datasets [51, 52] for building various state-of-the-art visual recognition models, current approaches leverage large amounts of videos available on multiple social media platforms. When specific language resources like automatically generated speech recognition captions are available in narrated video datasets such as How2 [53] or HowTo100M [12], an appropriate proxy task that leverages these resources is instead used. Such visual caption pairs have been widely used in self-supervised models in vision and language tasks recently [54, 55, 56, 57, 58, 59, 60, 61]. In other approaches like [39, 4, 62, 25, 63, 24], the need for these language transcripts is avoided by using just the corresponding raw speech signal. More recently, models that trained from scratch from the narrated video along with generated speech captions have also been successfully developed [1]. The three modalities naturally present in videos, the visual, audio, and language streams, are further integrated via a multimodal variant of this learning framework in [2]. Unlike these works, our goal in this paper is to learn a joint embedding in three modalitites for zero-shot multimodal downstream tasks where we create an embedding space which the features across different modalitites are directly comparable.

Contrastive Learning. A technique central to several state-of-the-art self-supervised representation learning approaches for images is instance-wise contrastive learning [64, 65]. In this paradigm, a model is trained to place samples extracted from the same instance, e.g., transforms or crops of an image, close to each other while pushing samples from different instances further apart. Given its similarity to noise contrastive estimation (NCE), where two samples are treated as a negative pair as long as they are drawn from different time segments, in MIL-NCE [1], the benefits of both multiple instance learning and NCE are combined. An advantage of this approach is that it now allows for compensation of misalignments inherently found in videos and corresponding text captions. One inherent drawback of the instance-wise contrastive learning described above is that it is agnostic to the inherent semantic similarity between the samples when positive and negative pairs

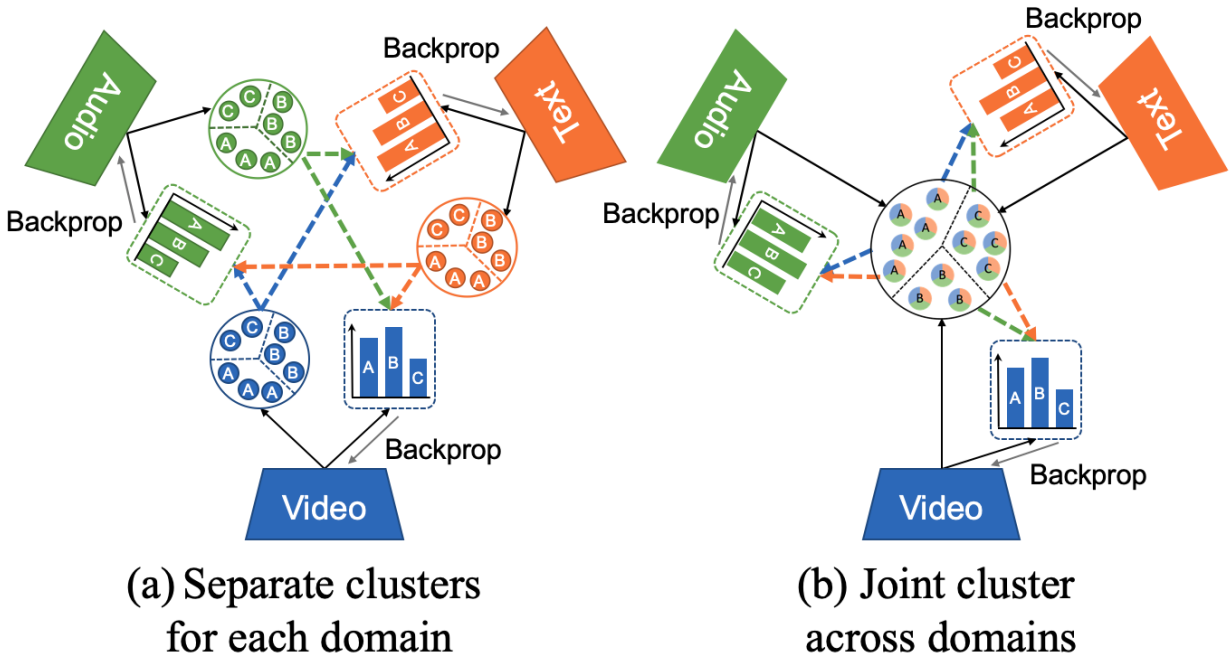


Figure 2.2: **Cross-domain Clustering vs. Joint Clustering.** (a) Previous methods such as XDC perform clustering at separate spaces and use pseudo-labels as supervision to other domains. (b) Our method performs clustering across features from different modalities in the joint space to learn multimodal clusters. Best viewed in color.

are constructed. In our work, we alleviate this problem by relaxing the instance level similarity across modalities to semantic level similarity by introducing a clustering component that learns semantic similarity among multimodal instances within the batch.

Deep Unsupervised Clustering. Given the high cost of computing all pairwise comparisons in a large dataset, instead of applying the contrastive learning paradigm discussed above on each individual instance, a more practical solution is to discriminate between groups of instances during training. This is done by first pre-training a model to derive suitable feature representations of the data in a simple cascaded approach. Keeping the representations fixed, a clustering algorithm is then used to group instances before the weights of the model are updated using the derived class assignments as supervision [31, 66]. In contrast, instead of keeping the clustering step independent of the representation learning phase, more recent techniques jointly learn visual embeddings and cluster assignments [45, 4, 46, 67]. While both these approaches can produce interpretable clustering results that benefit downstream tasks by integrating global information across the entire dataset,

running a clustering algorithm over a large data set slows down training. However, this issue can be addressed by performing the clustering in an online fashion [46]. These online models simultaneously learn to cluster and represent image data. To improve the performance of clustering, it is, however, also essential to leverage the correlated yet very complementary information available in the various modalities present in narrated videos [45]. To learn better feature extractors for audio and video, recent works, XDC [39] and SeLaVi [45] extend this clustering idea to the multimodal space. While these approaches focus on learning better feature extractors for each domain separately, our goal is to learn a joint multimodal embedding. As shown in Figure 2.2, these cross-domain clustering methods (left) create separate clusters and use cross-domain pseudo-labels as the supervision for each feature extractor. In contrast, our model (right) creates a common embedding space across all modalities and performs clustering jointly.

2.3 Learning to Cluster Multimodal Data

To effectively construct a *joint representation space* from unlabeled narrated videos, we start with n narrated video clips. Each video clip is associated with its corresponding visual representation, audio representation and text narration. Given this input, the joint embedding space is learned, where the embeddings of video clips with semantically similar visual, audio, and text content are close to each other and apart when the content is dissimilar, as illustrated in Figure 3.1.

Using the notation in [1], for each clip, let video $v \in \mathcal{V}$ denote its visual representation, $a \in \mathcal{A}$ represent its corresponding audio and $t \in \mathcal{T}$, its matching text narration generated using an automatic speech recognition (ASR) system. Given a set of n tuples of associated video, audio and text narrations $\{(v_i, a_i, t_i)\}_{i=1}^{n} \in (\mathcal{V} \times \mathcal{A} \times \mathcal{T})^n$, as shown in Figure 2.3 (a), we first construct three parametrized mappings that derive embedding representations from the original video, audio and text signals. Transform $f : \mathcal{V} \to \mathbb{R}^d$ derives a d-dimensional embedding representation $f(v) \in \mathbb{R}^d$ from a video clip v, transforms $g : \mathcal{A} \to \mathbb{R}^d$ and $h : \mathcal{T} \to \mathbb{R}^d$, produce similar d-dimensional audio and text embeddings: $g(a) = z \in \mathbb{R}^d$ and $h(t) \in \mathbb{R}^d$. In this work, f takes as input pre-extracted 2D and 3D features from a fixed-length clip, the input for g are log-mel spectrograms extracted from

the audio segments, and for h, we use a sentence based neural model that transforms a set of words into a single vector. More details about model architectures are in Section 2.4.

Next, we introduce three loss functions to guide and properly situate these embeddings in the joint embedding space. A contrastive loss L_{MMS} is used to ensure that the representations from each of the three modalities are comparable. A second clustering loss $L_{Cluster}$ encourages representations from semantically similar samples across all modalities to remain close in the learned embedding space. A third reconstruction loss $L_{Reconstruct}$ regularizes the multimodal common space features for more stable clustering training. The final model is trained to minimize sum of these losses.

$$L = L_{MMS} + L_{Cluster} + L_{Reconstruct} \tag{2.1}$$

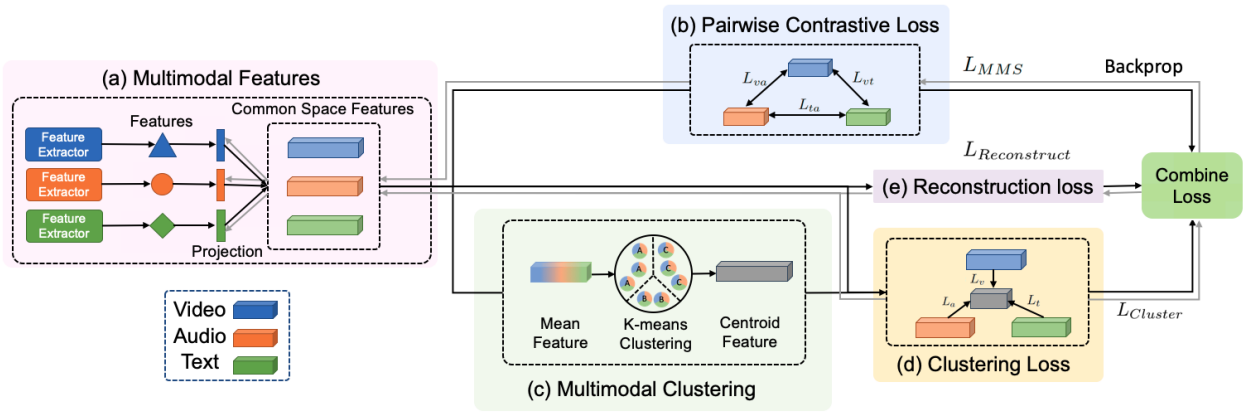


Figure 2.3: **Illustration of our proposed framework.** Our framework comprises four parts: (a) Extracting features from several modalities and projecting them into joint space. (b) Calculating contrastive loss pairwise to pull the features close across modalities. (c) Performing multimodal clustering across features from different domains in a batch. (d) Performing joint prediction across features to multimodal centroids to bring together semantically similar embeddings. (e) Reconstruction loss for regularization. Best viewed in color.

2.3.1 Contrastive Loss for Learning Joint Spaces

To learn a joint space for the three modalities, we compute a contrastive loss on all pairs of modalities, $(v,t),(t,a),(a,v)$, as shown in Figure 2.3 (b). This loss maximizes the similarity between representations corresponding to any two modalities from the same instance (video clip)

while minimizing the similarity of imposter pairs from the two modalities from one clip of video to another. In this work, we use the Masked Margin Softmax (MMS) function [68], which defines the similarity between representations from two modalities in terms of their learned embedding vectors' dot product within a batch B. Features from each of the three modalities $\{V, A, T\}$ are assembled for each batch. The total contrastive loss L_{MMS} is the sum of pairwise losses using each of the three modalities:

$$L_{MMS} = L_{ta} + L_{vt} + L_{va} \tag{2.2}$$

where L_{ta}, L_{vt}, L_{va} represent the loss associated with pairwise modalities (t, a), (v, t), (a, v) respectively. For a pair of modalities, for example the text and audio modalities, the individual loss L_{ta} is in turn given as:

$$L_{ta} = -\frac{1}{B}\sum_{i=1}^{B}\left[\left(\log\frac{e^{h(\mathbf{t}_i)\cdot g(\mathbf{a}_i)-\delta}}{e^{h(\mathbf{t}_i)\cdot g(\mathbf{a}_i)-\delta}+\sum\limits_{\substack{k=1\\k\neq i}}^{B} e^{h(\mathbf{t}_k^{imp})\cdot g(\mathbf{a}_i)}}\right)+\left(\log\frac{e^{h(\mathbf{t}_i)\cdot g(\mathbf{a}_i)-\delta}}{e^{h(\mathbf{t}_i)\cdot g(\mathbf{a}_i)-\delta}+\sum\limits_{\substack{j=1\\j\neq i}}^{B} e^{h(\mathbf{t}_i)\cdot g(\mathbf{a}_j^{imp})}}\right)\right] \tag{2.3}$$

where a_j^{imp} represents imposter pairs from two modalities that are sampled from a batch but do not co-occur. As can been seen in the L_{ta} case, this loss attempts to discriminate between positive or true embedding pairs and imposter or negative pairs within each batch. Using two separate parts, the space of positive and negative samples is enumerated separately: in one case, a given text sample is paired with various negative audio samples. In the second case, an audio sample is paired with various negative text samples. (i, j, k) are various indices of video clips in a given batch. δ is a margin hyperparameter that is empirically selected. By projecting all features to the same space and ensuring that their similarities are maximized pairwise, this formulation of the pairwise contrastive loss ensures that the features across different modalities are comparable.

2.3.2 Clustering Multimodal Features

To ensure that representations of semantically related instances are close in the learned joint multimodal space, in addition to contrastive loss described above, a self-supervised clustering step is included as part of the training process.

Online K-means clustering. We applied standard clustering algorithm k-means that takes a set of vectors as input, in our case, the features M produced by the fused multimodal feature:

$$M = (f(\mathbf{v}) + g(\mathbf{a}) + h(\mathbf{t}))/3 \tag{2.4}$$

where we take the mean over embeddings from three modalities to represent a multimodal instance. We cluster them into k distinct groups. More precisely, it outputs a $d \times k$ centroid matrix $C = \{\mu_1, .., \mu_k\}$ and the cluster assignments y_n of each multimodal instance n are defined by solving the following problem:

$$\min_{C \in \mathbb{R}^{d \times k}} \frac{1}{N} \sum_{n=1}^{N} \min_{y_n \in \{0,1\}^k} \|M_n - C y_n\|_2^2 \tag{2.5}$$

We then acquire a centroid matrix C^* and a set of assignments $(y_n^*)_{n \leq N}$. Unlike pseudo-labels-based methods [31] that only make use of the assignments (labels), we make use of the centroid matrix for semantic learning. To cover variant semantic information for clustering, we use features from the previous batches to gather sufficient instances for online learning.

Semantic centroid learning. To learn the features closer to its multimodal semantic centroids. We proposed to use the centroid as a contrastive loss reference target. This target pulls the features from three modalities closer to the centroid that is close to their multimodal instance feature M_n and pushes the features far away from the other centroid. For each modality, for example, the text modalities, the individual loss L_t is in turn given as:

$$L_t = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{e^{h(\mathbf{t}_i) \cdot \mu' - \delta}}{\sum_{k=1}^{K} e^{h(\mathbf{t}_i) \cdot \mu_k}} \tag{2.6}$$

where μ' is the nearest centroid for the multimodal instance feature M_i and μ'. We later sum over the loss from three modalities:

$$L_{Cluster} = L_v + L_a + L_t \tag{2.7}$$

In the end, the projected features learn to be closer to its centroid feature among the three and also learns to be closer in similar semantics.

Multimodal features reconstruction. Reconstruction can help in capturing features that are suppressed by contrastive learning/clustering [69]. In a video of *chopping onions*, with both the sound of chopping in the background as well as the speech/text with the word *onion* in the foreground, it is possible that contrastive learning/clustering will focus more on associating the video with either the sound (background) or the speech (foreground), but not both. We hypothesize that the reconstruction loss will force the capture of features from both background and foreground, which is important for retrieval/other downstream tasks. Reconstruction is also an auxiliary task that helps regularize training and improve generalization [70]. We performed a reconstruction loss on top of the common space features from three modalities to stabilize the feature training during clustering. For each modality, for example, the visual modalities, the individual loss $L_{v'}$ is in turn given as:

$$L_{v'} = -\frac{1}{B}\sum_{i=1}^{B}\|f'(\mathbf{v}) - f(\mathbf{v})\|^2 \tag{2.8}$$

where $f'(\mathbf{v})$ represented the reconstructed features by feeding $\mathbf{v}$ into two linear layers as encoder and decoder. We then sum the loss over each modality:

$$L_{Reconstruct} = L_{v'} + L_{a'} + L_{t'} \tag{2.9}$$

2.4 Experiments

2.4.1 Implementation details

For the visual branch of the proposed MCN model we follow [12] and use pre-trained 2D features from a ResNet-152 model [71] trained on ImageNet [72] to extract features at the rate of

one frame per second, along with pre-trained 3D features from a ResNeXt-101 model [73] trained on Kinetics [51] to obtain 1.5 features per second. The video clip features were computed by concatenating the 2D and 3D features into a 4096 dimension vector and max-pooling the features over time. For the audio branch of the network, we compute log-mel spectrograms and use a DAVEnet model [25] to extract audio features. For the textual branch, the feature extraction process proposed in [12] is adopted to extract text representations: a GoogleNews pre-trained Word2vec model [74] provides word embeddings, followed by a max-pooling over words in a given sentence to extract a sentence embedding. Note that all backbones are fixed, and they are not fine-tuned during training. Each feature extraction branch is followed by a separate fully-connected layer and a gated unit for projecting the features in a common embedding space. To allow for pairwise comparisons, features from each of the different modalities are set to be 4096-dimensional vectors. More details can be found in the supplement. We use an Adam optimizer [75] with a learning rate of 1e−4 and cosine learning rate schedule [76]. The model is trained for 30 epochs on four V100 GPUs over a period of about two days. Various hyperparameters in our experiments are set as follows: margin hyperparameter $\delta = 0.001$, and a batch size of $B = 4096$ video clips and cluster size is set to be 256.

2.4.2 Datasets

Training Dataset. Our models are trained on the HowTo100M [12] instructional video dataset, which contains 1.2M videos along with their corresponding audio that consists of speech and environmental sound and automatically generated speech transcriptions.

Downstream Datasets. The **YouCook2** [48] dataset contains 3.5K cooking instruction video clips with text descriptions collected from YouTube. Unlike Howto100m dataset, text descriptions in YouCook2 are human-annotated. The **MSR-VTT** [36] dataset contains 200K human annotated video clip-caption pairs on various topics. We use the same test set with 1K video clip-caption pairs constructed in [12] in our experiments. The **CrossTask** [49] dataset contains 2.7K instructional videos that cover various topics.The action steps and their order for each task were collected from

Method	Mod	Model	TR	YouCook2			MSRVTT		
				R@1	R@5	R@10	R@1	R@5	R@10
Random		-	-	0.03	0.15	0.3	0.01	0.05	0.1
Miech [12]	VT	R152+RX101	N	6.1	17.3	24.8	7.2	19.2	28.0
MDR [54]	VT	R152+RX101	N	-	-	-	8.0	21.3	29.3
MIL-NCE* [1]	VT	R152+RX101	N	8.1	23.3	32.3	8.4	23.2	32.4
MCN (ours)	VAT	R152+RX101	N	**18.1**	35.5	45.2	**10.5**	**25.2**	**33.8**
MDR [54]	VT	R152	N	-	-	-	8.4	22.0	30.4
ActBERT [61]	VT	R101+Res3D	N	9.6	26.7	38.0	8.6	23.4	33.1
SSB [59]	VT	R(2+1)D-34+R152	N	-	-	-	8.7	23.0	31.1
MMV FAC [2]	VAT	TSM-50x2	Y	11.7	33.4	45.4	9.3	23.0	31.1
MIL-NCE [1]	VT	I3D-G	Y	11.4	30.6	42.0	9.4	22.0	30.0
MIL-NCE [1]	VT	S3D-G	Y	15.1	**38.0**	**51.2**	9.9	24.0	32.4

Table 2.1: Comparison of text-to-video retrieval systems. Mod indicates modality used, where V: video, A: audio, T: text. TR indicates if a trainable backbone is used or not.

wikiHow articles with manual annotation for each frame. The **Mining Youtube** [50] dataset focuses on YouTube videos for five simple dishes.The test set contains 250 cooking videos, 50 of each task, that are densely annotated, *i.e.*, each frame is labeled with its respective action class.

2.4.3 Downstream Tasks

To demonstrate the effectiveness of the proposed model, we evaluate embeddings derived from the network in two downstream tasks: text-to-video retrieval and temporal action localization. We focus on the zero-shot task because we want to access the quality of the cross-modal semantic embedding that was learned during training. When performing retrieval using our model, we compare the query text features with the video and audio features by computing similarity for both and using the average. For action localization, we compute the same distance of the video-audio pair of each frame to each respective label embedding and are so able to align video frames to each of the provided action steps.

Text-to-Video Retrieval. The goal of this task is to retrieve the matching video from a pool of videos, given its ground truth text query description. The model is tested on two video description datasets and evaluated on recall metrics: R@1, R@5, R@10. These evaluations are used to

Method	Mod	Model	TR	CrossTask			MYT		
				Recall	IOD	IOU	Recall	IOD	IOU
CrossTask [49]	VT	R152+I3D	N	22.4	-	-	-	-	-
CrossTask [49]	VT	R152+I3D	N	31.6	-	-	-	-	-
Mining: GRU [50]	VT	TSN	N	-	-	-	-	14.5	7.8
Mining: MLP [50]	VT	TSN	N	-	-	-	-	19.2	9.8
Miech [12]	VT	R152+RX101	N	33.6	26.6	17.5	15.0	17.2	11.4
MIL-NCE* [1]	VT	R152+RX101	N	33.2	30.2	16.3	14.9	26.4	17.8
MCN (ours)	VAT	R152+RX101	N	35.1	**33.6**	**22.2**	**18.1**	**32.0**	**23.1**
ActBERT [61]	VT	R101+Res3D	N	37.1	-	-	-	-	-
ActBERT [61]	VT	+ Faster R-CNN	N	**41.4**	-	-	-	-	-
MIL-NCE [1]	VT	I3D-G	Y	36.4	-	-	-	-	-
MIL-NCE [1]	VT	S3D-G	Y	40.5	-	-	-	-	-

Table 2.2: Evaluation of temporal action localization systems.

demonstrate the effectiveness of the contrastive loss and learned joint embedding space across three modalities.

Text-to-Full Video Retrieval. The conventional text-to-video retrieval task attempts to match a caption (or ground-truth text query) to a single video clip. Since a single caption can refer to many individual clips within a dataset, this task is limiting. To this end, we propose the task of *text-to-full video retrieval* where the goal is to match a set of captions (or text queries) describing multiple parts of a video to an entire video. This is a more realistic task than single clip retrieval since various real-world applications require retrieving entire videos from complex textual queries. We evaluate on YouCook2 dataset with recall metrics: R@1, R@5, R@10.

Temporal action localization. We further evaluate our model on two temporal action localization tasks. The CrossTask [49] dataset considers the task of clip level action detection. Here, an unordered set of action labels is given for a set of clips of the same video, and clips have to be classified with the respective action labels. The performance is reported as recall and computed as a ratio of the correctly predicted clips over the total number of clips in the video as used in [49]. The MiningYoutube [50] dataset considers the task of frame-level temporal action segmentation.Here, each test video is provided together with the respective actions and their ordering, including the

background. The goal is to find the correct frame-wise segmentation of the video given the action order. We follow the inference procedure outlined in [50] to compute the alignment given our similarity input matrix. The dataset employs two evaluation metrics: intersection over detection (IoD) [77], defined as $\frac{G \cap D}{D}$: the ratio between the intersection of ground-truth action G and prediction D to prediction D, and the Jaccard index, which is an intersection over union (IoU) given as $\frac{G \cap D}{G \cup D}$.

2.4.4 Comparison with State-of-the-art Methods

Zero-shot Video Retrieval. We first examine the results of the text-to-video retrieval task on the YouCook2 and MSR-VTT datasets (Table 2.1). We compare only with baseline models that were not fine-tuned on the respective dataset for a fair comparison. To allow comparability between different approaches, we use a fixed visual feature extraction backbone as described in [12] whenever possible. For the baseline MIL-NCE* [1], we apply their training strategy on the same visual feature set we use, ResNet-152 (R152) and ResNeXt-101 (RX101) [12]. On YouCook2, our model significantly outperforms prior works on the same architecture and shows even competitive results compared to models with trainable visual backbone (TR). Our method also performs better than the other baselines on MSR-VTT. The gains are, however, not as significant as on YouCook2. We attribute this to the fact that neither the available audio nor the textual description is instructional in nature and, therefore, semantically further away from our training set.

Zero-shot Action Localization. We examine the action localization tasks on the CrossTask and the MiningYouTube dataset in Table 2.2. For CrossTask, given each frame in the video, we perform a zero-shot classification of the given labels and calculate the recall. In this zero-shot setting, the model computes video text similarity to localize action step labels similar to [12]. Our method outperforms state-of-the-art approaches for self-supervised learning [1, 12] and a fully supervised approach [49] especially in the IOU and IOD metrics, which also consider false-positive predictions from the background class as an action step. Approaches in [12] and MIL-NCE* [1] are directly comparable with our method since they use the same feature extractor as us. In contrast, MIL-NCE

[1] uses a stronger video backbone and [61] uses additional feature modalities such as region features along with a stronger language model. We also evaluate our model on the MiningYoutube [49] temporal action localization benchmark. Our method outperforms state-of-the-art approaches for both self-supervised [1, 12] and weakly supervised [50] learning. More settings, including data and computing resources for each model, are in the supplement.

Clustering Metrics. To better evaluate our learned features, we use the k-means clustering algorithm and calculate various clustering metrics based on ground-truth labels on the CrossTask [49] and MiningYouTube [50] tasks. In this case, the number of clusters k, also corresponds to the number of possible steps assigned to the temporal action localization task for each video during test time.

We follow the evaluation protocol and notations used in [45] and report performance based on the following standard clustering metrics: *normalized mutual information* (NMI) [78], *adjusted rand index* (ARI) [79], and *accuracy* (Acc). These results are obtained after matching the estimated k-means pseudo-labels to the ground truth targets using the Kuhn–Munkres/Hungarian algorithm [80]. We also report the *mean entropy per cluster* :

$$\langle H \rangle = \frac{1}{K} \sum_{k \in K} H(p(y|\hat{y}_k = k)), \tag{2.10}$$

where $\hat{y}$ corresponds to the psuedo-labels generated by clustering and y relates to the ground-truth labels. In this formulation $p(y|\hat{y}_k = k)$ denotes the distribution of ground-truth labels that fall in the generated clusters k, while $H(U)$ represents the entropy given as $-\sum_{i=1}^{|U|} P(i) \log(P(i))$. In ideal conditions, the perfect mean entropy will be zero.

We also report the the *mean maximal purity per cluster*,

$$\langle p_{\max} \rangle = \frac{1}{K} \sum_{k \in K} \max(p(y|\hat{y}_k = k)), \tag{2.11}$$

In ideal conditions, the perfect mean purity will be 100%.

By using the various metrics described above, the clustering result on MiningYoucook dataset

	CrossTask				
Method	**NMI** ↑	**ARI** ↑	**Acc.** ↑	⟨**H**⟩ ↓	⟨$\mathbf{p}_{\max}$⟩ ↑
Random	3.2	3.2	9.4	1.30	47.5
Miech *et al*. [12]	61.8	46.1	57.0	0.39	81.5
MIL-NCE* [1]	62.0	45.6	56.7	0.37	82.4
MCN (ours)	**65.5**	**48.5**	**57.6**	**0.34**	**83.8**

Table 2.3: Performance on clustering metrics on the CrossTask dataset evaluated by GT text annotations on video segments.

was shown in Table 2.3. The overall results show a similar pattern with the experiment shown in the main paper using CrossTask dataset. Results are shown in Table 2.3. It shows that our learned multimodal features are closer to the ground-truth distribution and have higher purity within the cluster.

2.4.5 Full Video Retrieval

To address the problem of full video retrieval from a set of captions, we divide each video into a set of clips, which are compared with the queries. We evaluate three different methods: In **majority vote over clip** predictions, we obtain the top-k predictions of each clip/caption pair as votes and select the video which has the majority of votes. For **majority vote over videos**, the maximal prediction over all the clips of a video is taken for each caption to obtain video/caption pairs. Then, the top-k of these predictions are selected as votes, and the video with the most votes is predicted. Lastly, our **caption averaging** method involves obtaining the maximal prediction over all the clips of a video is taken for each caption and then averaging over the set of captions in a query. This gives a single prediction for the entire video.

We examine the results of the text-to-full video retrieval task on the YouCook2 dataset (Table 2.5). Of the three methods to obtain full video predictions, the caption averaging achieves better results than both majority voting schemes. Furthermore, we find that our method outperforms prior works on this task with a 6.8% improvement on R@1. Since we obtain full video predictions, we also perform full-video classification on the CrossTask dataset using the set of sub-task labels as the

				YouCook2			
Method	Mod	Model	FT	R@1	R@5	R@10	Median R
Random		-	-	0.03	0.15	0.3	1678
Miech [12]	VT	R152+RX101	Y	8.2	24.5	35.3	24
MCN (ours)	VT	R152+RX101	Y	11.3	28.2	38.4	20
MCN (ours)	VAT	R152+RX101	Y	**28.2**	**53.0**	**63.7**	**5**

Table 2.4: Comparison of text-to-video retrieval systems on finetune setting. FT indicates if it is finetuned on the downstream dataset.

set of query captions, where we achieve a top-1 accuracy of 68.7%.

2.4.6 Zero-Shot Action Recognition

We also test our method's performance for the downstream task of zero-shot action recognition. For these experiments, we follow the evaluation protocol of [81] and test on the full UCF-101 and HMDB datasets. We present the top-1 and top-5 accuracies on both datasets in Table 2.6. Although MCN is trained using instructional videos, we find that the joint video/text space it learns is sufficient for the task of zero-shot action recognition. Furthermore, our method can be further improved by training on action-related videos; by removing various video categories - 'food and entertaining', 'computers and electronics', 'cars and other vehicles', 'home and garden', and 'health' and training on a subset of the HowTo100M dataset, we find MCN is able to achieve state-of-the-art Top-5 accuracy on both datasets. The baseline, [81], is a method designed specifically for zero-shot action recognition and is trained using labeled action videos from Kinetics-700, leading to strong top-1 accuracy.

2.4.7 Finetune results

We show our model's performance on the finetune setting in Table 2.4, which means we also train on an additional training set provided by the Youcook [48] dataset. Although the finetune setting, which requires ground-truth labels, isn't our main focus, we obtain significant improvement over the current baseline.

Method	Prediction	R@1	R@5	R@10
Random	-	0.23	1.15	2.32
MCN (ours)	MV-Clip	38.8	67.4	76.8
MCN (ours)	MV-Video	38.8	67.7	78.4
MCN (ours)	Caption Avg.	**53.4**	**75.0**	81.4
Miech *et al*. [12]	Caption Avg.	43.1	68.6	79.1
MIL-NCE* [1]	Caption Avg.	46.6	74.3	**83.7**

Table 2.5: Comparison of Text-to-Full Video retrieval systems on the YouCook2 dataset. The prediction column denotes the method used to obtain video-level predictions: majority vote over clips (MV-Clip), majority vote over videos (MV-Video), and caption averaging (Caption Avg.).

	UCF-101		HMDB	
Method	Top-1	Top-5	Top-1	Top-5
Brattoli *et al*. [81]	37.6	62.5	26.9	49.8
MCN (ours)	33.0	62.3	20.9	48.4
MCN-actions (ours)	33.9	63.7	22.5	51.5

Table 2.6: Zero-shot action recognition performance on the UCF-101 and HMDB datasets. MCN-actions is the MCN method, which has been “fine-tuned" on a subset of the HowTo100M dataset which contains action-related videos.

2.4.8 Ablation Studies

To better understand the contributions of various algorithmic design choices used to build the proposed MCN model, we perform a set of ablation studies on the following downstream tasks: YouCook2 R@10 (YR10), MSR-VTT R@10 (MR10), CrossTask average recall (CTR) and MiningYoutube IOU (MY-IOU). For each setting, we use the same feature extractor for three modalities as described in Sec 4.1 for a fair comparison. More ablations are in the supplement.

Selection on different losses. In our first set of experiments, we find the proposed clustering is crucial not only for clustering-related tasks but also for retrieval (MSR-VTT) tasks as shown in Table 2.9. This validates our hypothesis that semantically close instances should be clustered closely in the joint embedding space. Also, the selection of contrastive loss (MMS) shows better results in our model.

Different choices of clustering methods. We evaluate the performance of (1) Selection of different

Loss	YR10	MR10	CTR	MYT-IOU
NCE	39.2	33.5	33.9	21.5
MIL-NCE	40.0	33.0	33.7	21.1
MMS	43.7	32.9	34.3	22.1
MMS + Cluster	44.3	33.7	34.5	22.6
MMS + Cluster + Reconstruct	45.2	33.8	35.1	23.1

Table 2.7: Ablation study on different loss including the selection of contrastive learning loss, the additional clustering, and reconstruction loss.

Method	Target	Labels	YR10	MR10	CTR	MYT-IOU
Sinkhorn	Swap	hard	39.0	33.4	33.6	21.1
Sinkhorn	Swap	soft	41.8	33.9	34.5	22.1
Sinkhorn	Joint	hard	44.4	33.4	34.6	21.1
Sinkhorn	Joint	soft	43.6	32.4	34.1	21.6
K-means	Swap	hard	41.3	32.8	33.2	21.0
K-means	Joint	hard	44.3	33.1	34.6	21.4
K-means	Centroid	hard	45.2	33.8	35.1	23.1

Table 2.8: Ablation study on different clustering pipelines with various methods, loss prediction target, and label types.

clustering methods such as Sinkhorn clustering [4] and K-means [82]. (2) Different prediction targets such as using swap prediction, which uses the pseudo label of other modalities for prediction target as [46, 39]. Or using the mean feature pseudo label as a joint prediction for three modalities. Also, using the centroid of the cluster as the target. (3) Different prediction labels, including hard labels (one-hot) or soft labels (continuous). *Clustering method.* The goal of this analysis is to create various kinds of pseudo-labels as prediction targets. If a pseudo-label can be thought of as a certain semantic representation of a cluster, two instances that have the same pseudo-label, can then be considered as semantically similar. The K-means method follows the deep clustering [31] approach which utilizes K-means clustering to create pseudo labels as prediction targets. These targets are then used for single modality learning on ImageNet [83]. The Sinkhorn clustering method follows the SeLa [4] technique that utilized a trainable network to replace the K-means clustering for generating pseudo-labels. The method also applies an optimal transport sinkhorn algorithm [84] to guarantee uniform distribution over different cluster labels, which in turn prevents the learnable clustering network (2 layers MLP) from learning a degenerated solution. More details

Loss	YR10	MR10	CTR	MYT-IOU
Miech *et al*. [12]	24.8	28.0	33.6	11.4
MIL-NCE* [1]	40.0	33.0	33.7	21.1
MCN	45.2	33.8	35.1	23.1

Table 2.9: Ablation study on different loss including the selection of contrastive learning loss, the additional clustering, and reconstruction loss.

		YouCook2			MSRVTT		
Method	Mod	R@1	R@5	R@10	R@1	R@5	R@10
MMS	T→V	7.4	20.0	29.3	8.8	23.2	32.2
MIL-NCE*	T→V	8.1	23.3	32.3	8.4	23.2	32.4
Ours	T→V	8.6	24.1	33.4	9.6	23.4	32.1
MIL-NCE* + audio	A→V	16.2	36.6	43.7	13.2	28.4	33.3
Ours	A→V	19.4	41.3	50.9	14.8	30.1	39.0
NCE	T→VA	14.5	32.1	39.2	8.8	24.1	33.7
MIL-NCE* + audio	T→VA	15.1	31.9	40.0	9.0	23.3	33.0
MMS	T→VA	16.1	33.9	43.7	9.5	23.3	32.9
Ours	T→VA	18.1	35.5	45.2	10.5	25.2	33.8

Table 2.10: Comparison of retrieval across different modalitites.

of this sinkhorn clustering approach can be found in [4, 46]. *Prediction Target.* We investigate two sources of pseudo-labels as prediction targets. In the first approach, the **swap** prediction utilizes a pseudo-label created from a different domain as a prediction target. As shown in the yellow box of Figure 2.4 (c), pseudo-labels from the audio (orange) and text (green) domains are used as prediction targets for the visual feature (blue). This mechanism is similar to XDC [39] except that we perform this approach on projected features a in common space. In the **joint** prediction method, a mean feature from the features of three modalities is first computed as a multimodal feature representation. Later, its pseudo-label will be the prediction target for the three separate feature instances and will be used to guide the features to be close across modalities and semantics. As shown in Figure 2.4 (d), the pseudo-label of the mean feature is used as the prediction target for features of each of the three modalities. *Label type.* We have two kinds of labels: hard labels that represent discrete labels and soft labels that represent continuous, probabilistic labels. Since K-means assigns each instance

	YouCook2				MSRVTT			
Cluster size k	R@1	R@5	R@10	Median R	R@1	R@5	R@10	Median R
64	17.8	34.7	43.4	17	10.1	25.3	34.1	27
128	17.3	34.8	44.2	19	**10.5**	24.5	33.5	29
256	18.1	**35.5**	**45.2**	**16**	**10.5**	25.2	33.8	27
512	**18.3**	35.3	44.4	19	10.4	24.6	33.5	26.5
1024	17.9	34.6	43.5	17	9.4	**25.8**	**34.6**	**25**

Table 2.11: Comparison of text-to-video retrieval systems on different number of cluster size in K-means

to one of the centroids, it will only produce hard labels. The outputs from the Sinkhorn clustering are from a learnable network. We can use the softmax operator to transfer these outputs into probabilities over different labels (soft) or use the arg-max function to derive discrete labels (hard). When we perform soft-label prediction over the Sinkhorn pipeline as shown in (a), it will be similar to Swav [46], but we perform this over multiple modalities and treat the different modalities as a kind of data augmentation. As shown in Table 2.8, our method encourages each modality feature to move closer to the semantic centroid, which improves performance by explicitly encouraging semantically close features from different domains to cluster together.

Ablation of modalities. We perform ablation experiments on the use of modalities in Table 2.10. From these experiments we find audio information to be crucial in bridging the gap between video and text while learning a joint space across the three modalities. The improvement on MSR-VTT is not significant compared to Youcook2. We attribute this performance difference to the domain gap between the various datasets. Both HowTo100M and Youcook are based on instructional videos where the text modality has a strong correlation to the video and audio modalities. In HowTo100M, the text is based on ASR transcripts. In Youcook2 and MSR-VTT, the query texts are hand-annotated captions. While Youcook2 captions describe single cooking steps, MSR-VTT captions are general descriptions of the scene, with captions. These captions are often not close to instructional ASR and also less related to what is being said in the audio.

Different number of clusters Table 2.11 shows the results using different number of cluster sizes for K-means. The result shows similar performance across different datasets and evaluation metrics.

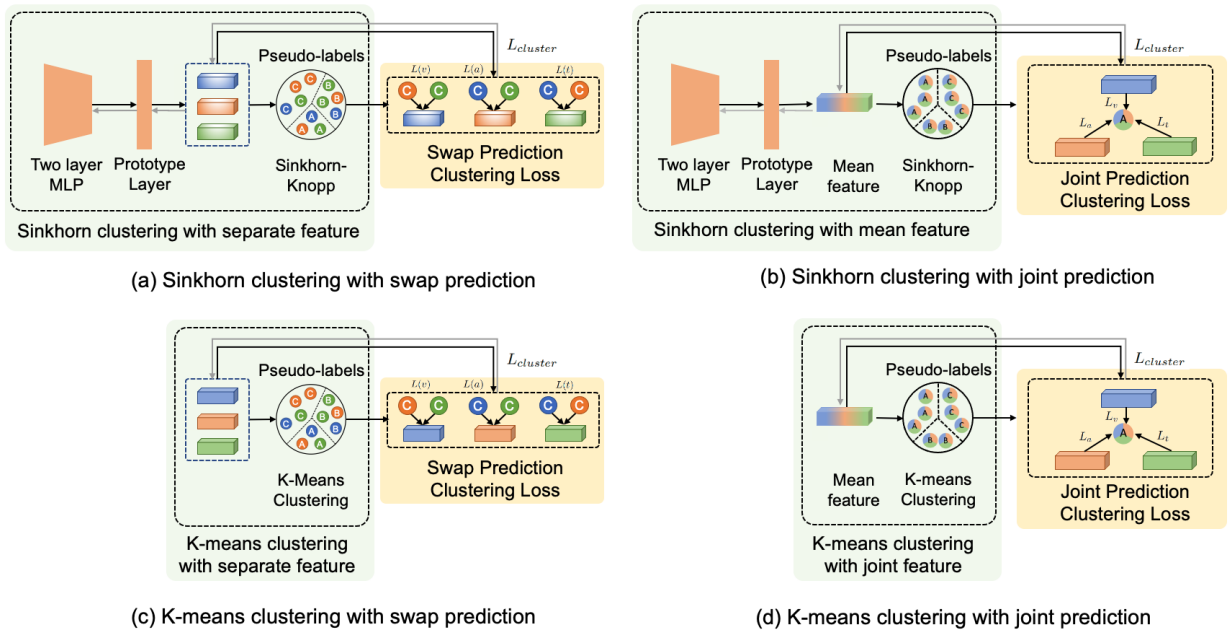


Figure 2.4: Comparison of different clustering pipelines. We investigate different clustering pipelines in replace of the clustering loss in our main paper. (a) Performs a sinkhorn clustering folloing a swap prediction. The loss was calculated between the clustered features and pseudo labels. (b) Replaces the swap prediction to joint prediction by performing the clustering on the mean feature. The loss was calculated by the mean pseudo label and the projected feature in Figure 3a. (c) Performs K-means along with swap prediction. (d) Performs K-means on the mean features and performs joint prediction.

2.4.9 Qualitative Analysis

We perform a qualitative analysis with the model's ability to do zero-shot text-to-video retrieval shown in Figure 2.5. Given an open-vocabulary caption, our model can retrieve the correct corresponding video segment. We also visualize the efficacy of using multimodal embeddings (concatenated video and audio representations) over using only visual embeddings. Representations from the CrossTask dataset are visualized using t-SNE plots. We observe that with multimodal features as Figure 2.6 (b), semantically related instances (based on ground truth classes) tend to be more tightly related than uni-modal visual features trained from contrastive loss (a) that appear more spread out. Also, multimodal features are clearly more separable for different actions. We also perform a qualitative analysis with the model's temporal action localization results on the MiningYoutube task. One interesting observation is shown in Figure 2.7. We observed that our

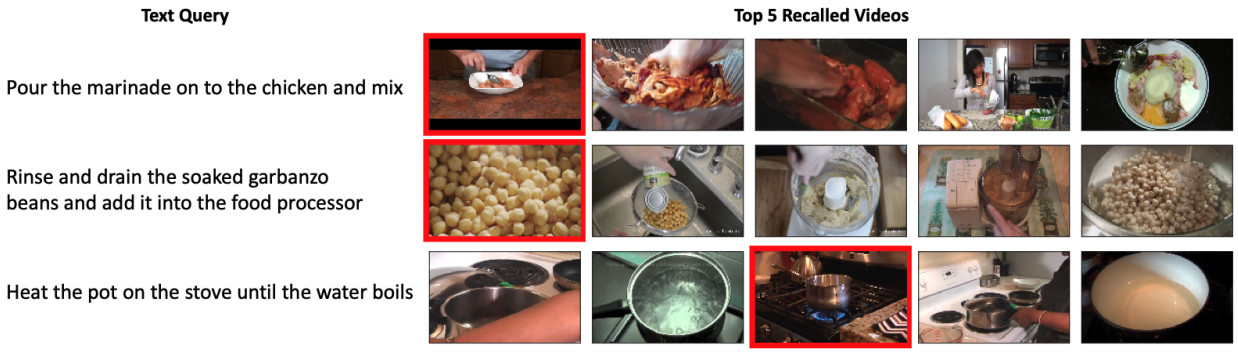


Figure 2.5: Qualitative results for the text-to-video retrieval task on YouCook2. Top-ranked clips show a high similarity to the described task as well as among each other without being too visually similar.

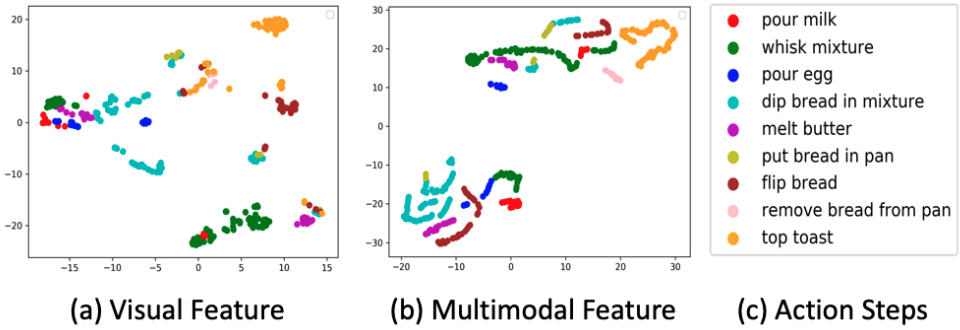


Figure 2.6: t-SNE visualizations on the CrossTask dataset for the task of "Make French Toast". Best viewed in color.

model performs well in distinguishing action steps from the background scenes. We attribute this improvement to the proposed clustering component, which we observe has separated the background frames from various action classes. Background class instances are often placed as outliers with respect to the various action step clusters.

2.5 Summary

In this chapter, we have developed a novel self-supervised multimodal clustering network that learns a common embedding space by processing local (via a contrastive loss) and global (via a clustering loss) semantic relationships present in multimodal data. The multimodal clustering network is trained on a large corpus of narrated videos without any manual annotations. Our extensive

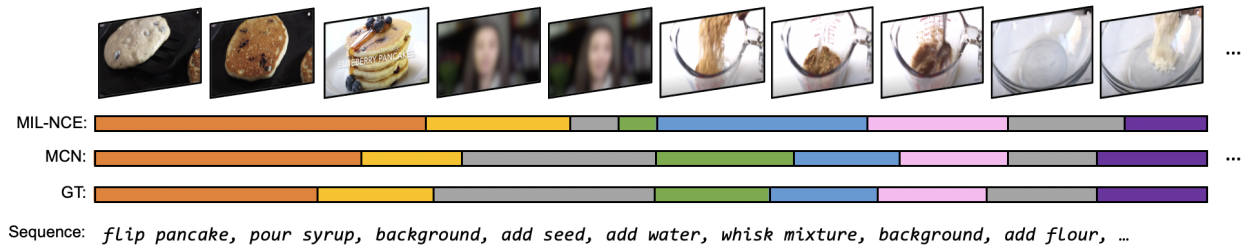


Figure 2.7: Temporal action localization example from the first minute of the video "Vegan Blueberry Quinoa Pancakes" in the MiningYouTube dataset. Given the video and the action step sequence, the goal is to align the step temporal boundaries.

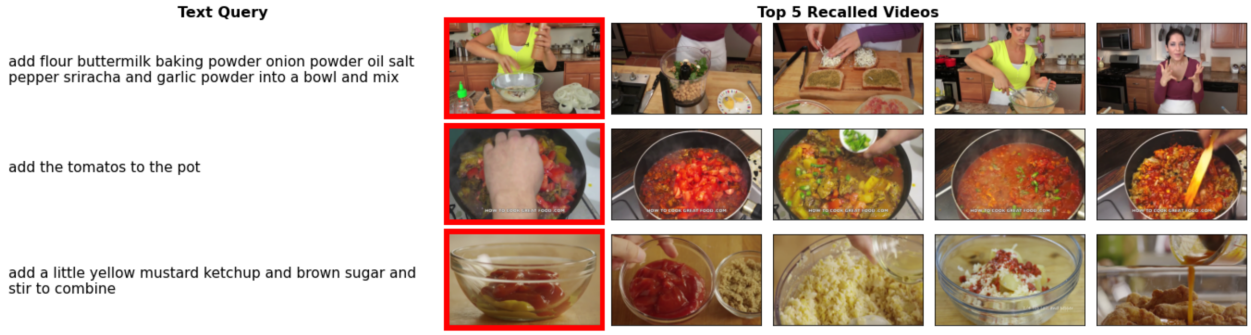


Figure 2.8: Text-to-video retrieval examples. The retrieved video clips show a similar pattern.

experiments on multiple datasets show that creating a joint video-audio-language embedding space with a clustering loss is essential for self-supervised learning of good video representations. Our approach can be extended to more modalities such as optical flow or sentiment features and applied to other multimodal datasets for learning joint representation spaces without human annotation.

Chapter 3: Self-supervised Spatio Temporal Grounding

3.1 Introduction

The problem of multimodal self-supervised learning from videos has gained increased research interest in the last few years. Starting from the initial approach by Miech et al. [12] that leveraged video and respective automatic speech recognition (ASR) text captions from large-scale instructional videos for training, recent approaches learn high-level representations using complementary modalities such as video, ASR text, and audio without human annotation [1, 2, 3, 85]. Beyond achieving state-of-the-art performance on various tasks after fine-tuning, the resulting models also show great capabilities for zero-shot tasks such as cross-modal video retrieval or classification, and especially allow for zero-shot temporal action segmentation and detection [49, 50, 86, 85, 87]. They are thus able to detect events in videos without labeled training data and based on referential expressions only.

At the same time, another line of research emerged focusing on the problem of label-free spatial grounding from multimodal data, mainly image-caption [88, 89, 90, 91, 92, 93] or video-caption pairs [94, 95]. Here, the goal is to correctly localize a referential expression in an image or in each video frame, e.g., via a bounding box or a heatmap. The underlying assumption is that the evaluated expression is visible in the image or all video frames. Those methods are thus not optimized to detect whether an event is present in a video.

The following work aims to bring together those two streams of work to address the task of spatio-temporal action grounding from multimodal supervision in untrimmed videos. Namely, we propose a grounding approach that uses video-text pairs based on ASR transcripts in instructional videos and learns the spatial representation of free-text events as well as their temporal extent as shown in Figure 3.1. To this end, we leverage two different representations of the visual data, a

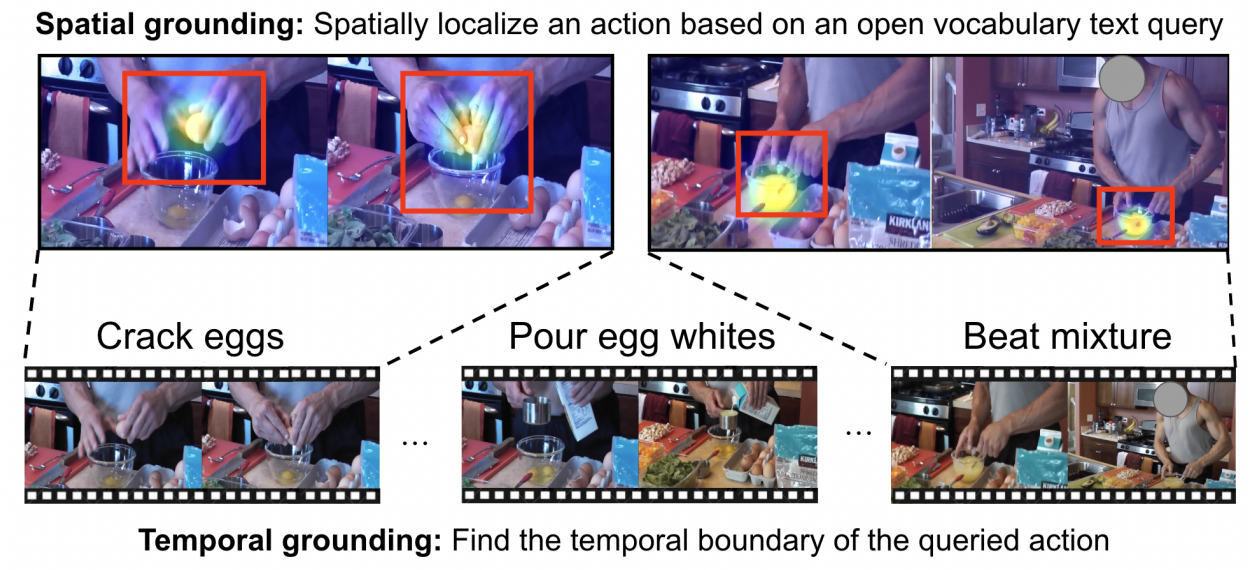


Figure 3.1: **Spatio-temporal grounding in untrimmed videos:** Given an input video, we perform spatio-temporal grounding using an action description such as "crack egg" as a query. The model needs to localize both the action's temporal boundary and spatial region in the long untrimmed video. We visualize the heat-map from the annotation points as well as derived bounding boxes.

global feature representation based on full-frame information to define the temporal extent of an event, and a local representation based on frame-wise grid features for spatial localization.

The motivation for this separation is that the local representation captures the spatial correlations between vision and text input, but this can be too fine-grained to learn a holistic representation of the frame at the same time. The global representation can thus be assumed to capture a more compact, aggregated view compared to local data and thus to provide a more reliable cue for the task of temporal localization. However, compared to the clean image-caption setup of most spatial grounding methods, the ASR text is not precisely aligned with the described activities since actions and events may occur before or after being described or being scattered over multiple frames [1, 96]. Therefore, we need to refine selection of frames and capture those useful for training. To this end, we look for frames that actually match the vocabulary of the respective text, leveraging a selection strategy by Sinkhorn optimal transport [84]. This allows us to train a model that can localize action instructions and semantic concepts in space and time within videos without labeling supervision.

To evaluate the task of spatio-temporal grounding in untrimmed videos, we annotated a new benchmark based on the existing Mining YouTube dataset [50] and extended it with spatio-temporal

localization information. This setup differs from other datasets [94, 97, 98] in two ways: first, we specifically focus on the spatial-temporal grounding of actions compared to e.g. objects, second, the dense annotations allow us to benchmark action grounding in long, untrimmed videos compared to existing temporally pre-clipped data. We provide a labeling for 512 different event types throughout the entire video including background class, resulting in over $5K$ spatio-temporal event annotations as shown in Figure 3.1.

To evaluate the proposed method as well as the new benchmark, we train the system on the HowTo100M dataset [12] and compare it to state-of-the-art methods for spatial and temporal grounding, as well as combined spatio-temporal grounding tasks. It shows that existing methods are usually doing well in one of the two aspects, spatial or temporal grounding, but not on both, while the proposed method is able to detect actions spatially and temporally based on semantic concepts without label annotation during training. We summarize our contributions as follows[1]:
(1) We propose the new task of spatio-temporal grounding in untrimmed videos based on weakly aligned multimodal supervision. (2) To address this task, we leverage a combination of global representation learning to encode temporal information and local representations to learn the spatio-temporal extent of actions in instructional videos. (3) We provide a new benchmark and annotations to evaluate this challenging problem on real-world instructional video data.

3.2 Related Work

Multimodal Self-supervised Learning. Multimodal Self-supervised methods aim at learning data representations by leveraging a large amount of unlabeled data with multiple co-occurrences of modalities. Early work [99, 100] proposed to project images and text into a joint visual-language embedding space where image and text embeddings of semantically-similar pairs are close. Following this idea, CLIP [7] learned representations leveraging 400 million image-text pairs publicly available on the internet. Other approaches [101, 102, 103, 104] exploit correspondence between the visual and the audio streams to learn representations from unlabeled videos. Miech

[1]We will make the code and the annotations publicly available.

et al. [12] trained an effective video-language embedding space by introducing the HowTo100M dataset with 1.2 million instructional videos collected from YouTube paired with text descriptions from ASR. Learning representations from text, visual, and audio modalities has also been studied [41, 105]. In this context, [2, 3, 24, 85] recently showed that using videos without annotation enables an effective multimodal embedding space via contrastive learning.

Spatial Video Grounding. Spatial video grounding, as a special category of multimodal self-supervised learning, aims to identify spatial locations corresponding to text descriptions. This task is mostly studied in the context of video object detection in supervised learning [106, 107, 108] or weakly supervised learning scenarios without temporal tracking detection capability [98]. Among object grounding benchmarks, the YouCook2-BoundingBox [98] dataset provides bounding box annotations for the visible objects in the YouCook2 [38] dataset of cooking videos. Recent work proposed the YouCook-Interactions dataset [94] together with an approach for the spatial grounding of objects and actions with multimodal self-supervision from HowTo100M videos. All of those works focus on spatial grounding only and assume that the video is temporally clipped with respect to the grounding phrase.

Temporal Video Grounding. Temporal video grounding aims to determine the set of consecutive frames corresponding to a text query in an untrimmed video [109, 110, 111, 112, 113]. In the context of actions, temporal boundaries of action instances are predicted. Previous work can be categorized into proposal-based and proposal-free approaches. Proposal-based approaches employ a propose-and-rank pipeline framework to localize the temporal boundaries of the target segment [114, 115, 116, 117, 118, 119, 120, 121]. These methods are computationally expensive and rely on proposal quality. Among proposal free methods, [122, 123] uses attention-based grounding and [124, 125] proposed reinforcement learning for regressing start and end times of target video segments. However, the majority of methods are trained on curated datasets with temporal boundary annotations in fully supervised settings and lack spatial localization ability [126, 127].

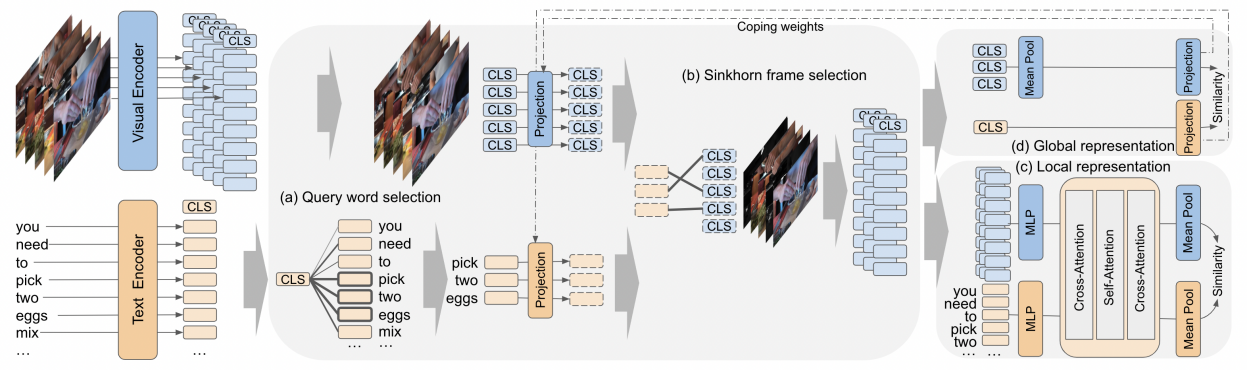


Figure 3.2: **Spatio-temporal localization model.** (a) We first select most relevant words on the sentence using the [CLS] sentence embedding. (b) We then select frames with possible groundable objects by matching the selected, projected word features with respective frames features utilizing the Sinkhorn optimal transport. (c) Based on the selected frames, we learn a local representation to ground the action description to the spatial region and (d) a global representation to allow for temporal localization.

3.3 Method

The goal of our method is to construct two representation spaces from unlabeled videos, a local and a global one. We start with narrated video clips, each associated with a corresponding visual representation and text narration. Given this input, the joint embedding spaces for both global and local representations are learned via contrastive loss by bringing the embeddings of semantically similar visual and text content closer together for both representations. For each clip $\mathcal{X} = \{\mathcal{V}, \mathcal{S}\}$, let $\mathcal{V}$ stand for the video clip and $\mathcal{S}$ for the text narration sentence generated by the automatic speech recognition (ASR) system. Each video clip $\mathcal{V}$ consists of $T \times N$ spatio-temporal tokens $\{v_{t,n}\}$, where $t \in \{1, ..., T\}$ represents the number of frames in the video and $n \in \{1, ..., N\}$ represents the number of spatial grid region tokens or features in a frame. The text sentence $\mathcal{S}$ consists of K words $\{s_1, ..., s_K\}$. We represent localized features by the tokens from each modality, and the global features $\{V, S\}$ are acquired either by mean-pooling over the local features or by using the classifier token from the transformer as in [7]. We learn transformations $f : V \rightarrow \mathbb{R}^d$ to a d-dimensional representation $f(V) \in \mathbb{R}^d$ from the global representation V, and $g : S \rightarrow \mathbb{R}^d$, to produce similar d-dimensional text global embeddings: $g(S) \in \mathbb{R}^d$. Similar to $\{f, g\}$, we note $\{f', g'\}$ to be the transform for localized features, where local features $\{v, s\}$ are also projected as d-dimensional representations. In this work, f takes as input S3D [128] or CLIP transformer [7] features from

a fixed-length clip, and the inputs for g are from a sentence-based neural model that transforms a set of words into a single vector. In our method, a global contrastive loss $\mathcal{L}_{Global}$ is used to ensure that the representations from each of the modalities at the global level are comparable. A second localized attention contrastive loss $\mathcal{L}_{Local}$ encourages representations from finer granularity, e.g., spatial regions and words, to be close in the embedding space.

3.3.1 Representations guided frame sampling

Learning representations from multimodal self-supervision is challenging since the narration is very likely to be not aligned with the video [96, 1], which is one of the key differences between weakly supervised vision-caption grounding and multimodal self-supervised grounding. Motivated by this, we pursue a frame selection strategy when sampling videos while training to learn better object grounding and temporal alignment. We start from a longer sequence U, where $U > T$, which includes the video frames before and after the ASR boundaries that might contain actions or objects in the sentence. Our goal is to find T frames out of the U frames that are most relevant to the actions and objects in the sentence $\mathcal{S}$. To this end, we start by selecting T words from the sentence $\mathcal{S}$ and utilize each word as the query to pick the T relevant frames in the untrimmed video. Hence, the selected frames contain certain object/action concepts from each word. All words are ranked by the feature similarity between each word in the sentence and the global sentence-level feature, e.g. the [CLS] token in case of a transformer model, selecting the top T words that best represent the sentence for our grounding target as shown in Figure 4.2(a). We assign the selected T words to T out of U frames by formalizing it as an optimal transport problem utilizing the Sinkhorn-Knopp algorithm [84].

Optimal transport for word to frame assignment. To acquire the optimal assignment from word features to video frames, we compute an assignment matrix $\mathbf{Q}$ from each video and ASR pair. This cross model *self-labelling* mechanism is applied to create labels $\mathbf{Q}$ from the projected cross-model similarity $\mathbf{P}$ between word tokens and each video frame, where $\mathbf{P} = g(\mathcal{S}) \bigotimes f(\mathcal{V}_t)^T \in \mathbb{R}^{T \times U}$. We reuse the projected layer from the global representation in Figure 4.2(d), which will project

multimodal features into a common space for feature similarity calculation. To ensure that the word-to-region assignment contains more diversity instead of just identical assignments for each word, we add a constraint that requires label assignments to be equally distributed across various spatio-temporal regions [4, 46]. This is achieved by restricting $\mathbf{Q}_\mathbf{v}$ to a *transportation polytope* $\mathcal{Q}_v$:

$$\mathcal{Q} = \left\{\mathbf{Q} \in \mathbb{R}_+^{U\times T} \mid \mathbf{Q}\mathbf{1}_T = \tfrac{1}{U}\mathbf{1}_U, \mathbf{Q}^\top \mathbf{1}_U = \tfrac{1}{T}\mathbf{1}_T\right\}, \tag{3.1}$$

which enforces the pseudo-assignment distribution $\mathbf{Q}$ to assign equal marginal probability to each of the U frames, instead of converging to a single frame. The vector $\mathbf{1}_U$ represents one vector with dimension $U \times 1$.

The next goal is to enforce this *transportation polytope* $\mathcal{Q}$. A solution for $\mathbf{Q}$ is now computed using the optimal transport Sinkhorn-Knopp algorithm [46, 84] as shown in Figure 4.2 (b). The Sinkhorn-Knopp algorithm also normalizes the distribution of $\mathbf{P}$ as:

$$\mathbf{Q} = \mathrm{Diag}(\alpha)\exp\left(\frac{\mathbf{P}}{\varepsilon}\right)\mathrm{Diag}(\beta), \tag{3.2}$$

where α and β are scaling vectors that restrict $\mathbf{Q}$ to have a uniform distribution across region assignment. ε is a parameter that controls the smoothness of the mapping [46].

We later select the T frames in the corresponding assignment from the words for further training. Note that our selection part $\mathbf{P}$ is from a trainable projection. While acquiring a better word-to-region projection during training, we hypothesize that the frame selection also benefits. We ablate the respective frame selection strategy in Table 3.6a.

3.3.2 Local representations for spatial localization

To capture multimodal interaction with finer granularity, we apply the widely used attention mechanism to learn the projection between tokenized features as shown in Figure 4.2(c). We extract spatio-temporal region features v_{tn} from the video. Also, we extract word features s_k which represents the feature from word k. All tokenized features are projected through a linear layer. To

compute attention between the tokenized features, we stacked two cross-modal attention layers with a self-attention layer in the middle, as illustrated in Figure 4.2 (c). Cross-modal attention is computed similar to the standard attention mechanism [129]. Given a spatio-temporal token v_{tn} from a video, we compute the attention score to all of the words s_k, where $k \in \{1, ..., K\}$ in the ASR sentence $\mathcal{S}$ by $\alpha_{tnk} = \frac{\exp(e_{tnk})}{\sum_{k=1}^{K} \exp(e_{tnk})}$ in the same video clip, where $e_{tnk} = cosine(v_{tn}, s_k)$. We then acquire a contextual video token feature $\bar{v}_{tn} = \sum_{k=1}^{K} \alpha_{tnk} s_k$, which encoded text contexual information. Note that the contextual vector is represented by aggregating the representations from the other modality. We follow the standard self-attention computation [26] where K, Q, V represent the features for the keys, queries, and values represented as:

$$Attn(K, Q, V) = \text{softmax}\left(\frac{(Q^\top K)}{\sqrt{d_k}}\right) V \tag{3.3}$$

where d_k is the dimension of the key. In our case, we feed each contextual features $\{\bar{v}_{tn}$,$\bar{s}_k\}$ right after the first cross-attention layer to be the K, Q, V to acquire its self-attended representation. The localized attention model was trained using the contrastive loss.To represent the video clip $\mathcal{V}$ and ASR sentence $\mathcal{S}$, we mean-pool over the spatio-temporal tokens in video $\bar{V} = \frac{1}{TN} \sum_{r=1}^{TN} \bar{v}_r$, and words $\bar{S} = \frac{1}{K} \sum_{k=1}^{K} \bar{s}_k$ respectively. Let $\left(\bar{V}^{(l)}, \bar{S}^{(l)}\right)$ be the l-th training example pair. We adapt the Noise Contrastive Estimation (NCE) loss [30] and the localized attention losses $\mathcal{L}_{Local}$:

$$\begin{aligned} \mathcal{L}_{Local} = -\frac{1}{B} \sum_{l=1}^{B} \Bigg[\Bigg(\log \frac{e^{\bar{V}_l \cdot \bar{S}_l - \delta}}{e^{\bar{V}_l \cdot \bar{S}_l - \delta} + \sum\limits_{\substack{k=1 \\ k \neq l}}^{B} e^{\bar{V}_k^{imp} \cdot \bar{S}_l}} \Bigg) \\ + \Bigg(\log \frac{e^{\bar{V}_l \cdot \bar{S}_l - \delta}}{e^{\bar{V}_l \cdot \bar{S}_l - \delta} + \sum\limits_{\substack{k=1 \\ k \neq l}}^{B} e^{\bar{V}_l \cdot \bar{S}_k^{imp}})} \Bigg) \Bigg] \end{aligned} \tag{3.4}$$

where B stands for the batch. $\bar{V}_k^{imp}$ and $\bar{S}_k^{imp}$ represent imposter samples, and δ is a margin hyperparameter.

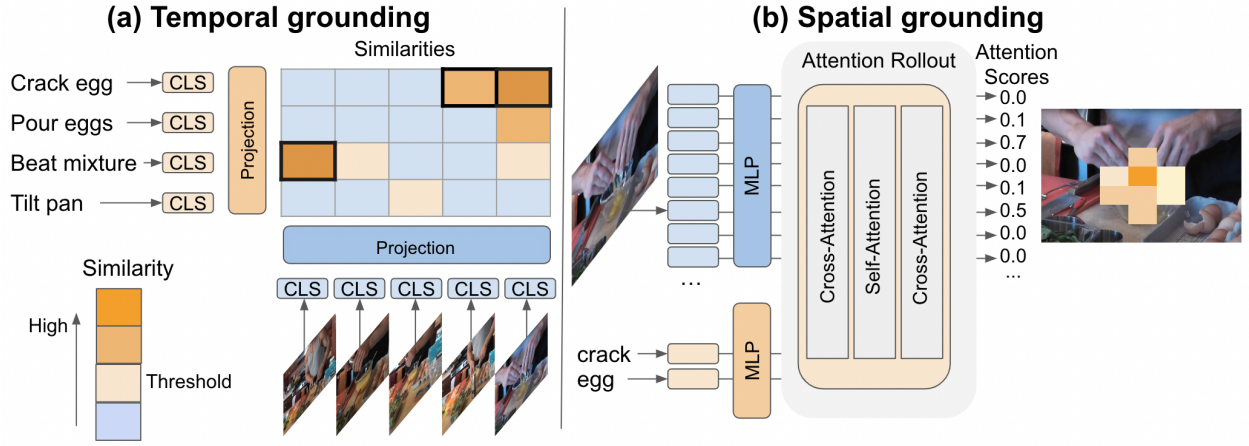


Figure 3.3: **Spatio-temporal inference.** We utilize both temporal and spatial representation during inferencing for spatio-temporal grounding. We start from predicting the action boundary on untrimmed videos. Spatial grounding is then performed using the predicted label as query to find corresponding regions.

3.3.3 Learning multimodal global representations

We learn to project the global representation of a whole video clip and a sentence by contrastive loss as shown in Figure 4.2(d). This loss pulls the representations of the two modalities from the same instance closer while pushing the imposter modality pairs sampled from different videos further away. We use the NCE loss function [30]. The global contrastive loss $\mathcal{L}_{Global}$ follows the same formulation as Equation 3.4 while uses the global representations V and S, which is the [CLS] tokens from both modalities in this case, instead of the local representations. Projecting the global features to the same space ensures that the features across different modalities are comparable. Since global representations encode information from the entire video, it is essential in encoding temporal information for the later downstream tasks. The final model is optimized by the sum of both losses.

Inference for spatio-temporal grounding. To perform spatio-temporal grounding on untrimmed videos, we start from temporal action description as shown in Figure 3.3. Given the possible action descriptions and an untrimmed video, we perform feature similarity matching using the global representation ([CLS] token) per frame with a threshold τ to filter backgrounds. We pick the class with the largest similarity score per frame. Later, we use the predicted label and feed it into the

local representation branch to compute spatial grounding. We follow attention rollout [94, 130] to compute feature similarity between visual tokens and text tokens through the cross-attention and self-attention. In the end, we acquire an attention heatmap for later downstream evaluation.

3.4 GroundingYoutube Benchmark

Current downstream datasets either provide spatial [94, 95] or temporal annotation [49, 50, 86]. These datasets do does not provide the opportunity to evaluate both aspects, spatial and temporal grounding, together. We therefore extend one of the current benchmarks, MiningYouTube [50], which already provides dense temporal annotations and we annotate video clips in the dataset with spatial information.

Annotating the spatio-temporal extent of actions can be challenging as there is no clear visible outline as in object annotation, nor is there a unique signal to indicate the temporal begin and end points. Similarly, grounding systems do not usually produce pixel-exact bounding boxes but rather indicate regions of interest. Detector-free spatial grounding models [89] address this fuzziness by relying on pointing game accuracy, thus only using the center point of the heat map for evaluation. Lending on this idea, annotators were asked to mark the presumed center point of the action. Compared to bounding boxes, center point annotation can be advantageous because annotators are not visually distracted by object outlines, so it is more likely that the most important region will be selected. We capture five annotations per frame, resulting in a density-based heat map.

Starting from 5,091 clips showing one of the 512 action classes, we adopt the methodology used for temporal action localization developed in [131] and label one frame per second, resulting in $26,987$ frames. We annotated all frames with five repeats per image, resulting in $134,935$ point labels in total. Following the previous evaluation setting using bounding boxes [132], we get the union of all annotated points in a single frame with an additional distance for constructing the bounding box.

3.4.1 GroundingYoutube Annotation

The data annotation was divided into three phases: During *Phase I* (Sec. 3.4.2, a graphical user interface (UI) and the task description were developed. In *Phase II*, the dataset was given to the annotators to generate the key points (Sec. 3.4.2). In *Phase III*, a manual quality control step was performed (Sec. 3.4.3).

3.4.2 Development of the graphical user interface and task description

The annotation of a large amount of data is often one of the most expensive aspects of a machine learning pipeline design, which is why the annotation time per datum should be kept as short as possible. There are two points that can be optimized, (1) the training or the task "message" for the annotators and (2) the graphical user interface by minimizing interaction times.

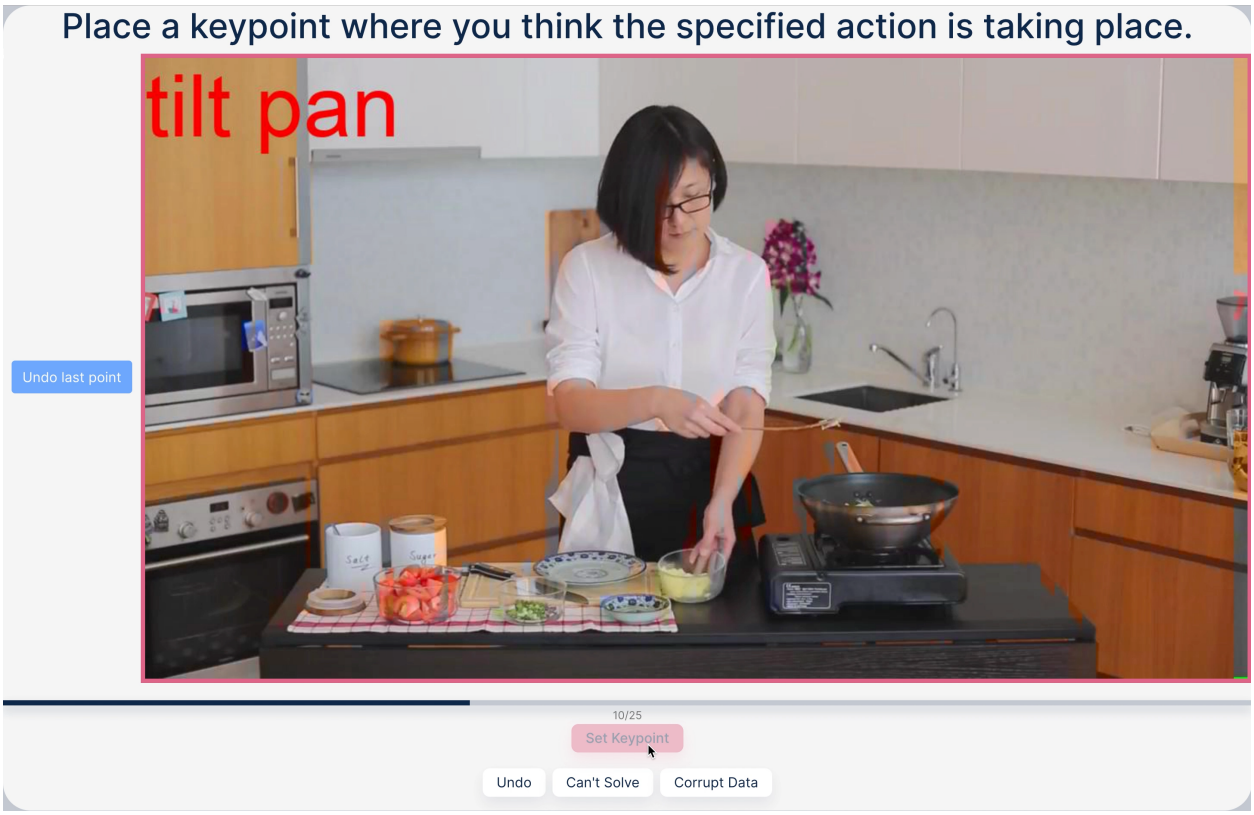


Figure 3.4: **A screenshot of our simplified annotation interface.** On the top, the annotation task is described in simple and short words to save reading time. To make interacting with the UI as intuitive as possible, actions are limited to simple button clicks and setting the key point by clicking on the image.

While tasks are usually formulated in such a way that no ambiguities arise, i.e. all possible edge cases are somehow covered, and simple words are used, in this case, we made a conscious decision to choose questions as short as possible, and that would give the annotator room for interpretation. We did this because it was hard to predict where people would actually locate actions in images. We also created a 1 min 30 sec long user training video where we demonstrate the task using exemplary keypoint annotations and explain how to use the UI.

Our annotation UI was designed with a special focus to keep it as intuitive as possible and reducing the interaction time. Our UI only provided five functionalities (set/unset a keypoint, undo the last image, image can't be solved, and image is corrupt) which were clearly described in text buttons (see Figure 3.4). Further, to reduce the cognitive load of our workers, our images were labeled in the form of work packages, each containing 25 images. Hence, we could ensure that completing a task would take no longer than 6 minutes.

The annotation of all 26, 987 images was performed with five distinct repeats per image, resulting in 134, 935 labels in total. All labels were generated by 13 professional annotators in total, which took them 5s in average per image. However, it should be noted that the number of images where an annotator placed a keypoint differs along all the workers (see Figure 3.5) and that the vast majority of all images have been answered by five annotators only. Examples are shown in Figure 3.6.

During the annotation, professional annotators were given a short instruction video at the beginning and then asked to click on the center of the given action without additional instructions. They were further free to choose "can't answer" if they could not locate the action, e.g., at the beginning and end of the clip. Thus, the number of available key points per image differs, and we choose majority voting to determine whether an action is present, resulting in new, refined temporal boundaries compared to the original annotation.

We found that the point-wise annotation resulted in roughly three distinct patterns, which depend on the captured scenario, as shown in Figure 3.7. In the case of half portrait or even wider shots in Figure 3.7a, annotations are highly locally centered. We further found that in some cases, the point annotation can also represent the flow of the action, e.g., pouring oil in Figure 3.7b, or even split

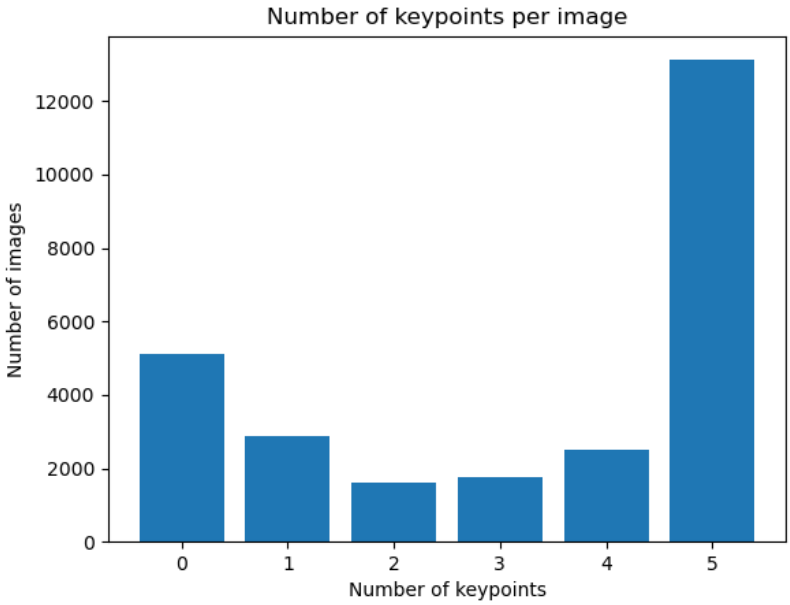
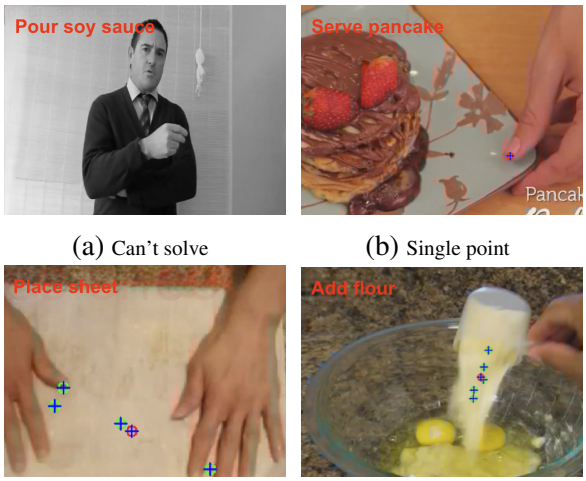


Figure 3.5: **Number of keypoints per image**. It can be seen that 48% of the data has all 5 key points and 19% has not a single annotation

into two separate clusters in Figure 3.7c.

3.4.3 Quality control

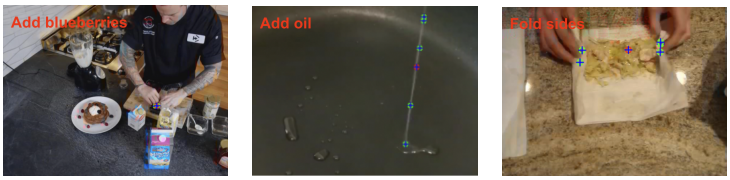
Since the label quality of the datasets used is a critical factor in the performance of machine learning models, we verified the correctness of a subset of our images using an experienced annotation specialist for 1,026 randomly selected frames. To evaluate the data quality, we evaluate the agreement between the annotation specialist and the annotations provided by the annotators. To this end, we considered an annotation as a false positive if three annotators or more have set a key point, although no action can be seen in the image, and as a false negative if three annotators or more have not set a key point, even though an action can be seen in the image. The entire sample was assessed using these criteria, with the specialist disagreeing with the annotators in only a total of 1.1% ± 3% (FP: 0.7% ± 3%, FN: 0.4% ± 3%). We also found that annotations significantly diverted in terms of spread. Namely, wider shots tend to be highly centered, whereas zooming in together with the usage of larger objects such as a pan or a spatula results in more widespread key



(a) Can't solve (b) Single point

(c) Four points (d) Five points

Figure 3.6: **Sample annotations**. The purple point represents the center point of the annotations in the frame. 48% of the data has all 5 key points, and 19% has not had a single annotation.



(a) Wider shot (b) Flow action (c) Split to two

Figure 3.7: **Example of keypoint annotations under different conditions.**

points. We also analyzed how often those cases occur and found that 14.0% of the selected frames show a widespread pattern.

Sample size calculation

To this end, we first needed a representative subset of N_S images of our data. We calculated the required sample size based on the following two formulas:

$$N_0 = \frac{z^2}{\epsilon^2} \cdot p \cdot (1 - p) \tag{3.5}$$

where α is the confidence interval, p the expected probability of the appearance of a quality aspect (e.g., widespread answers), $epsilon$ is the accepted error margin, and $Q(\alpha)$ is the percent point function of a normal distribution and $z = Q(1 - \frac{\alpha}{2})$.

As N_0 would be the required sample size for an infinitely large population, we applied the finite

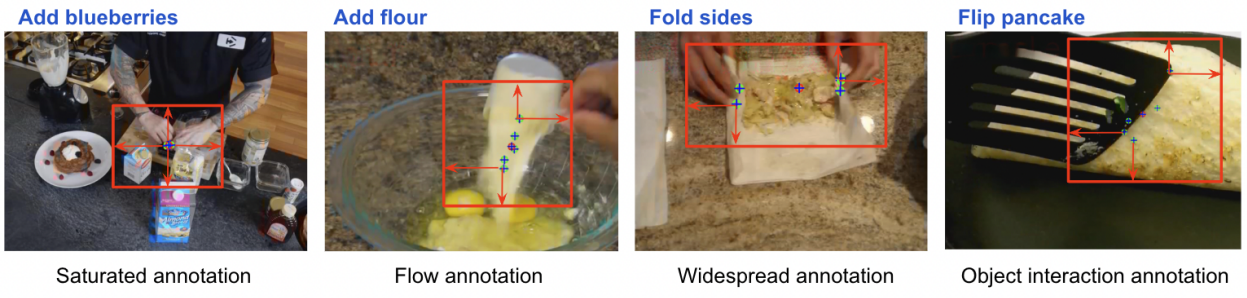


Figure 3.8: **Visualization on automatic bounding box generation from points.**

population factor that results from sampling without replacement from a finite population.

$$N_S = \frac{(N_0 \cdot N)}{N_0 + (N-1)} \tag{3.6}$$

where N is the total number of images.

We set $\alpha = 95\%$, $epsilon = 3\%$, and our sample size of $N = 26,987$. As the probability of the quality aspect is unknown, we set $p = 50\%$, which resulted in $1,026$ being checked for quality control.

3.4.4 Dataset usage for evaluation

Bounding box generation: For evaluation purposes, we get the union of all annotated points in a single frame with additional distance respect to the height H and width W as shown in Figure 3.8. We manually check the auto-generated bounding boxes and adjust the bounding box when needed.

3.5 Experiments

3.5.1 Experimental setup

In our method, we tested on CLIP[7] and S3D-word2vec[1] models. The following is the experiment setup and inferencing details.

CLIP models. For visual and text backbone, we use the pretrained weights from CLIP [7] with transformer ViT-B/32 and fix the encoder. Both the visual and text encoder has an embedding size

Method	Backbone	Data	Super.	Mod.	YouCook-Inter	GroundingYoutube		V-HICO		Daly	
					Acc	Acc	mAP	Acc	mAP	Acc	mAP
MIL-NCE [1]	S3D-word2vec	HT100M	Self	VT	23.67	27.45	8.21	12.65	11.23	13.84	24.23
CoMMA* [94]	S3D-word2vec	HT200K	Self	VT	48.63	47.68	23.38	40.97	21.45	54.48	33.39
Ours	S3D-word2vec	HT200K	Self	VT	**53.98**	**60.62**	**44.93**	**44.32**	**24.31**	**66.35**	**45.93**
CLIP [7]	CLIP	HT200K	Weak	IT	14.10	12.50	3.49	29.23	12.51	18.02	27.28
CoMMA† [94]	CLIP	HT200K	Self	VT	52.65	47.56	36.42	55.20	34.54	61.06	44.37
GLIP [91]	Swin-L	Cap24M	Weak	IT	52.84	53.62	24.73	**66.05**	**41.17**	-	-
Ours	CLIP	HT200K	Self	VT	**57.10**	**55.49**	**43.12**	60.71	39.28	**70.08**	**50.56**

Table 3.1: **Video spatial grounding**. We evaluate using pointing game accuracy and mean average precision. Models learning global representations (MIL-NCE, CLIP) don't perform well on localization tasks, while our model outperforms other grounding methods. We listed CNN-based methods on top and transfomer-based methods at the bottom. Our method generalized well on both architectures. (Mod. indicates the modality used, where V: video, I: image, T: text. Super. indicates supervision.)

Method	Backbone	Data	Super.	IoU	IoD
Mining: MLP [1]	TSM	MiningYT	Weak	9.80	19.20
CoMMA* [94]	S3D-word2vec	HT200K	Self	2.05	5.63
MIL-NCE [1]	S3D-word2vec	HT200K	Self	18.69	26.74
Ours	S3D-word2vec	HT100M	Self	19.18	27.65
Ours	CLIP	HT200K	Self	**19.88**	**28.50**

Table 3.2: **Action step alignment on MiningYoutube.** Spatial-focused model CoMMA is not trained to learn temporal representations, which results in lower performance. Our model is trained along with global representation and frame selection strategy, which achieved better temporal localization.

of 512. The sample rate is 1 frame per second with $T = 8$ frames in total. We set the number of possible frames U as described in Section 3.1 to be selected to 16. The decision choice experiments are in Table 3.4. We set a batch size of $B = 64$ video clips.

S3D-word2vec models. For the video backbone, we follow [94] and use S3D initialized by MIL-NCE on HowTo100M [1] at the rate of 5 frames per second and fix the video encoder. The global video clip features were max-pooled over time and projected into embeddings of dimension 512. For the text feature, we follow [12] to use GoogleNews pre-trained word2vec model [74] and max-pooling over words in a given sentence to acquire the text global feature. We set a batch size of $B = 96$ video clips.

With both backbone settings, we use an Adam optimizer [75] with a learning rate of 1e−4. For

inferencing in temporal grounding, as shown in Figure 3a, we first normalize the global feature for video and text, and we set a temporal threshold $\theta = 0.5$ to determine the background label. In spatial grounding, we acquire an attention heatmap using the attention rollout [130] described in Section 3.3. We set a spatial threshold $\tau = 0.01$ to create the mask, as shown in Figure 3b. The decision choice is shown in Table 3.3. In the ablation study Table 4e, we add the audio modality into training. For the audio branch, we compute log-mel spectrograms and use a DAVEnet model [25] initialized by MCN on HowTo100M [85] to extract audio features. We extend the global and local loss pairs from VT to VT, VA, and AT following [87]. The model is trained for 10 epochs on 4 V100 GPUs, which takes about one day.

Treshold	Backbone	0.1	0.05	0.01	0.005	0.001
CoMMA*	S3D-word2vec	0.76	0.90	0.93	0.91	0.86
Ours	S3D-word2vec	15.35	15.88	16.22	16.34	16.12
CoMMA†	CLIP	0.88	0.92	0.99	0.94	0.91
Ours	CLIP	15.93	16.33	17.10	17.05	16.24

Table 3.3: **Threshold for attention score on GroundingYoutube *mAP*@0.4**

GLIP baseline inference. In spatial grounding, we are given a text query and need to localize it in the frame. GLIP predicts multiple bounding boxes corresponding to the text query. We select the predicted bounding box with the highest confidence score as the prediction result. We use the center point of the predicted bounding box for the pointing game evaluation as the model prediction. For *mAP* evaluation, we use the predicted bounding box to compute IoU with the ground truth bounding box. In spatio-temporal grounding, we input all possible action description labels as candidates similar to Figure 3a. We pick the class with the highest confidence score as the predicted label. If the model made no prediction, we would predict as "background". The spatial inference is the same as the spatial grounding setting.

MIL-NCE, CLIP baseline inference. Both models are trained for global representations where we can compute sentence-to-video frame similarity to perform the temporal grounding for Figure 3a. We follow the same process as our method for temporal grounding. For spatial grounding, we compute sentence-to-region feature similarity. Both visual backbones produce a 7x7 grid feature.

We normalize the sentence, and region features, then select a spatial threshold $\tau = 0.5$ to create the mask for *mAP* evaluation.

# of frames	12	16	20	24	28
YouCook-Inter	56.3	57.1	56.8	56.7	55.9
GroundingYoutube	16.4	17.1	17.0	16.8	16.1

Table 3.4: **Ablation of # of frames used for selection**

3.5.2 Datasets

Training Data: We follow [94] and train on a subset of the **HowTo100M dataset**, comprising 250K randomly selected video clips from the *Food and Entertaining* category.

Downstream Datasets: YouCook2-Interaction: To evaluate the spatial grounding abilities of our system, we use the recently proposed YouCook2-Interaction dataset [94], an extension of a subset of the YouCook2 dataset [38] for cooking instruction, which provides bounding boxes for 6K selected frames. The bounding boxes usually comprise the hand and the tool mentioned in the respective sentence-wise annotation. **MiningYoutube:** To evaluate the temporal grounding abilities, we leverage the MiningYoutube [50] dataset, as it provides temporal annotation and, similar to YouCook-Interaction, is limited to the domain of cooking instruction videos. The dataset features 250 full instructional videos, which are annotated with 512 action classes and temporal boundary information. Here, temporal alignment, the task of finding the right temporal boundaries given the sequences of actions, is used during evaluation to relax the task of temporal detection. **GroundingYoutube**: To finally address the problem of spatio-temporal grounding, we leverage the extended version of the MiningYoutube dataset with dense spatio-temporal location information for 512 verb-noun phrases. All occurrences of the specific phrase in the test video are hence annotated, allowing us to evaluate spatio-temporal grounding in full untrimmed videos. To further benchmark on different domains besides cooking, we test spatial grounding on the **V-HICO** dataset [133] with 6.5k videos with human-object interaction bounding boxes annotations that have been

semi-automatically curated from sentence captions, as well as on the **Daly** action dataset [134], featuring videos consisting of daily actions such as "brushing teeth" and "cleaning windows".

3.5.3 Baseline methods

We compare the proposed system with other multimodal self-supervised methods to evaluate the approach and related data annotation. Namely, we choose MIL-NCE [1] as the standard baseline for this task, which utilizes S3D[128] and word2vec[74] to project two modalities into a common space. We include CoMMA [94] as the best performing model for spatial representations in self-supervised learning. We noted as CoMMA* to represent the model uses weights shared by the author[2]. CLIP [7] is an image-text model trained with transformer architecture on image caption pairs which shows great results on multimodal video tasks [135]. We further apply CLIP as the backbone and train with [94] to construct CoMMA†. GLIP[91] is a state-of-the-art model to combine the grounding task and object detection work. We include GLIP as a baseline to represent the SOTA image-text transformer model trained on large-scale image caption pairs. Such supervision is consider as weak supervision since the captions were written by humans and are well-aligned with the image, which is a cleaner supervision than untrimmed video-ASR signal. For the models using S3D[128] visual backbones, we follow [94] using the pre-trained weights from MIL-NCE [1] for initialization. For the models using S3D-word2vec features, we follow [1] to use the max-pooled word embedding to represent the sentence (global representation) since there is no [CLS] token. Also, the sentence feature is used for the query word selection instead of the [CLS] token. We used the mean-pooled S3D spatio-temporal features to represent the global representation of the video following the S3D architecture [128]. For CLIP[7] backbones, we use the pretrained transformer ViT-B/32. More implementation details and experimental settings are in the supplementary.

[2]We thank the authors for providing code and weights.

Method	Backbone	DataSet	Supervision	Modality	GroundingYoutube						
					IoU+Point	mAP					
						0.1	0.2	0.3	0.4	0.5	0.1:0.5
MIL-NCE [1]	S3D-word2vec	HT100M	Self	VT	4.67	33.94	25.16	12.65	3.42	0.41	15.11
CoMMA* [94]	S3D-word2vec	HT200K	Self	VT	1.02	2.18	1.72	1.11	0.93	0.37	1.26
Ours	S3D-word2vec	HT200K	Self	VT	**9.12**	**42.70**	**35.49**	**25.16**	**16.22**	**10.05**	**25.92**
CLIP [7]	CLIP	HT200K	Weak	IT	3.59	29.54	22.15	9.16	2.48	0.39	12.74
CoMMA†	CLIP	HT100M	Self	VT	1.68	3.51	2.32	1.88	0.99	0.40	1.82
GLIP [91]	Swin-L	Cap24M	Weak	IT	1.24	2.83	2.10	1.52	0.96	0.37	1.56
Ours	CLIP	HT100M	Self	VT	**10.09**	**42.81**	**36.05**	**25.84**	**17.10**	**11.35**	**26.63**

Table 3.5: **Spatio-temporal localization on full videos**. Since our model learns global representations encoding global information and spatial correspondences across modalities, it achieves a better performance in spatio-temporal evaluation compared to models trained on only spatial or temporal grounding. (V: video, I: image, T: text.)

3.5.4 Comparison with state-of-the-art methods

We considered the following downstream tasks to evaluate spatio-temporal grounding abilities of various models:

(i) **Spatial grounding:** The task is given a text query description to localize the corresponding region in the trimmed video. We use Youcook-Interaction, GroundingYoutube, V-HICO, and Daly for evaluation. This task is evaluated using the **pointing game accuracy**. Given the query text and video, we compute the attention heatmap on the video as described in Figure 3.3(b). If the highest attention similarity score lies in the ground truth bounding box, the result counts as a "hit" and counts as "miss" otherwise. The final accuracy is calculated as a ratio between hits to the total number of predictions $\frac{\text{\# hits}}{\text{\# hits+\# misses}}$. We report the mean average precision **(mAP)** following the settings from V-HICO [133] of the Known Object setting. Given a human-object interaction category as the text query, we aim to localize the spatial location in the video frame. The predicted human and object location is counted as correct if their Intersection over-Union (IoU) with ground truth human and object bounding boxes is larger than 0.3. Since we do not use any bounding box proposal tools or supervision, we create an attention heatmap as described in Figure 3.3(b) to create a mask for IoU computation. We follow [133] and compute the mAP over all verb-object classes. As shown in Table 3.1, models trained with global representations such as MIL-NCE and CLIP

were not able to localize the text description compared to models learning local representations such as CoMMA, GLIP, and our approach. In the cooking domain, we achieved the best result among all methods. In the open domain, such as V-HICO and Daly, our method also achieved competitive results, showing the generalizability of our model to other domains. We attribute this to the transformer architecture in the text branch inheriting knowledge from the open domain during large scale training, while in contrast the model's performance using word2vec dropped in these datasets. In the Daly dataset, the classes are verbs, which are not detectable by the object-focused model GLIP.

(ii) **Temporal grounding:** In this setting, each test video is provided together with the respective actions and their ordering, including the background. The goal is then to find the correct frame-wise segmentation of the video, given the action order. We follow the inference procedure outlined in [50] to compute the alignment given our similarity input matrix. The dataset employs two evaluation metrics: intersection over detection (IoD), defined as $\frac{G \cap D}{D}$ the ratio between the intersection of ground-truth action G and prediction D to prediction D, and the Jaccard index, which is an intersection over union (IoU) given as $\frac{G \cap D}{G \cup D}$. As shown in Table 3.2, we found the global representations played an important role in representing temporal information.

(iii) **Spatio-temporal grounding in untrimmed video:** In the main evaluation on our annotated GroundingYoutube dataset, we combined the spatial and temporal grounding as before to form the spatio-temporal evaluation. The entire video and the respective action instructions were provided. The model needs to localize each action step in temporal (start-time/end-time) and spatial (location in the video) as described in Figure 3.3. We evaluate in two metrics: **IoU + Pointing game** combines the evaluation setting from the spatial grounding and temporal grounding metrics. The frame in the video is counted as correct when the predicted class is correct. Also, given the predicted class as a query, the maximum point of the heatmap lies within the desired bounding box. We then compute the IoU over all the predictions with the GT to acquire the final score. We also follow previous spatio-temporal evaluations, which compute **video mAP** [131], where we set IoU threshold between GT and predicted spatio-temporal tubes. A prediction is counted as correct when it surpasses the

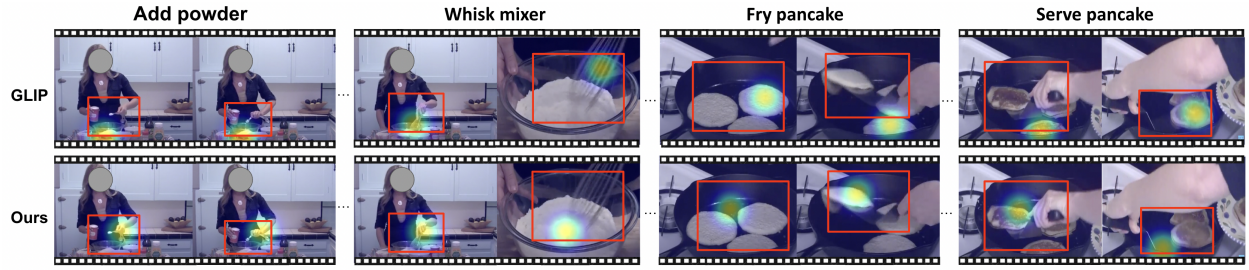


Figure 3.9: Visualization on GroundingYoutube dataset. The red box is the annotation and heatmap is the prediction from the model.

IoU threshold. We then compute the mean over the Average precision over all classes. We form a 3D prediction mask following Figure 3.3 and compute IoU between our 3D heatmap and 3D tube. As shown in Table 3.5, our method outperforms the other baselines by a significant margin. The result demonstrates our model's ability to incorporate both global (temporal) and local (spatial) representations to perform untrimmed video spatio-temporal action localization. Models designed for trimmed videos [94] or trained with aligned image-text [91] failed to capture the temporal dynamics, while models without specific loss designs for spatial grounding [1, 7] were not able to ground the action in the correct region.

3.5.5 Ablation study

We perform a set of ablation studies on the spatial grounding using YouCook-Interaction pointing game and spatio-temporal grounding on GroundingYoutube using mAP with IoU@0.4.

Frame selection strategy. We perform an ablation on the frame selection strategies in Figure 4.2(b). In Table 3.6a, *None* directly uses the ASR boundary as our video training data. *Global* uses the sentence feature [CLS] token as the query to rank the top T similar frames as the selected frames for training. *Local* uses the selected words in Figure 4.2(a) as a query and selects the frames with closest feature distance. We have shown that selecting frames based on possible groundable objects improves overall performance, leading to better supervision.

Number of frames for training. We tested different video lengths T used for training. As shown in Table 4.8b, selecting less frames for training significantly causes the performance to drop. We hypothesize that not only does the model fail to capture the temporal dynamics with less frames, but

Frame Sampling	None	Global	Local	Sinkhorn
YouCook-Inter	55.5	55.4	56.3	57.1
GroundingYoutube	15.1	15.7	15.6	17.1

(a) Effect of different frame selection strategy.

Frame length	1	4	8	16	24
YouCook-Inter	31.1	48.2	55.5	57.1	56.1
GroundingYoutube	5.2	9.5	16.1	17.1	16.5

(b) Effect of # video frames used for training

Loss	w/o global loss	w/o local loss	w/ both loss
YouCook-Inter	54.3	32.5	57.1
GroundingYoutube	5.7	7.6	17.1

(c) Effect of global and local loss functions

Dataset	HT200k	HT370k	HT100M
YouCook-Inter	57.1	56.8	57.0
GroundingYoutube	17.1	17.3	17.4

(d) Effect of different training dataset

Train/test supervision	VT/VT	VAT/VT	VAT/VAT
YouCook-Inter	53.9	53.6	53.8
GroundingYoutube	16.2	16.8	17.0

(e) Effect of audio supervision in training and testing

Table 3.6: **Ablations for training:** We isolate the effects of our training components. We find that **(a)** frames selected by the Sinkhorn strategy result in better supervision for grounding. **(b)** increasing the video length during training improves the performance, but decreases when the video length is too long since it includes irrelevant signals. **(c)** both loss contributes to final loss, the existence of global loss helps localization task itself. **(d)** training with more data improves slightly or no improve. **(e)** training with audio help us learn temporal information.

loses some frames with groundable objects in the sentence while training. We also found that when the number of frames increases, more irrelevant frames might be selected during training, which decreases the performance.

Global and local loss. As mentioned in the spatio-temporal evaluation, global and local features both contribute significantly to the final grounding result. We test the model by ablating out each loss. In the spatial grounding result, as shown in Table 4.8c, adding the global loss improves the ground performance. This results also shows that spatial grounding benefits from global representation

learning. In the spatio-temporal setting, the performance without a global or local loss outperforms other baselines.

Dataset for training. As mentioned in Section 3.5.2, we trained models with data with food categories. In Table 4.8d, we also tested our model trained with a larger set of food and entertaining called HowTo370K used in [96]. The full set of HowTo100M contains a total of 1M long videos, which is five times the size of our dataset. We found training with our 200K videos reaches similar performance with much less training hours.

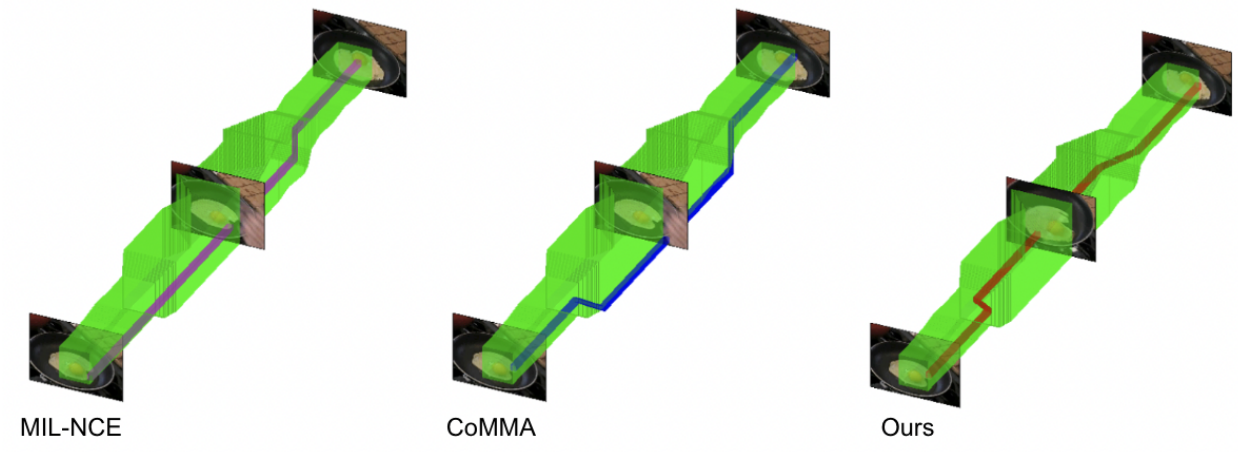
Affect of audio in training and testing. Unlike text which describes a discrete concept as a target to ground, audio serves as a continuous representation that is highly relevant to the temporal information. For example, we can determine an action started when we hear a "cracking" sound. In Table 3.6e, we tested our model using the additional audio modality by expanding our architecture and loss from VT to VAT. We found when training and testing with audio, the spatio-temporal result increases while the spatial-only result remains the same. This validates our assumption that audio contributes more to temporal understanding. When we trained on audio and tested without audio, the performance increases over the VT model, showing that the audio serves as useful supervision for better video/text representations. More details are presented in the supplement.

3.5.6 Design choices

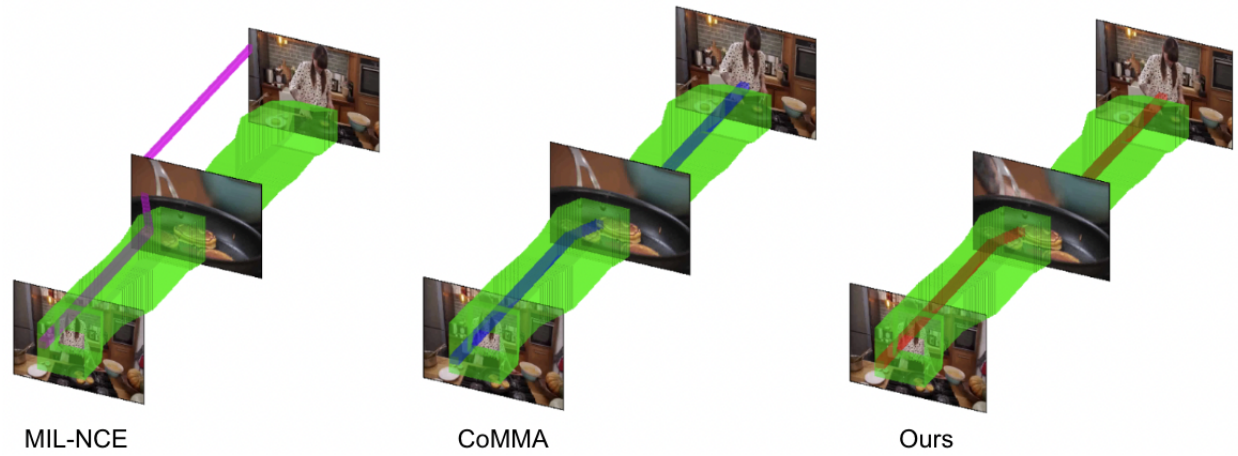
Frames used for selection. As shown in Table 3.4, we perform an ablation study on the number of candidate frames U used for training. We found that selecting 16 frames achieved the best performance, balancing the useful video information in training. Also, it doesn't include too many irrelevant concepts that are different from the action/object in the ASR sentence.

Threshold for attention mask. As shown in Figure 3.3, we apply a threshold to create a mask from the result of attention rollout. Note that this threshold τ is not a hyperparameter that affects the training or the model but simply serves as a means to an end to compute the mAP scores. We didn't systematically optimize this threshold but instead chose it as giving the most plausible qualitative results. We tested different thresholds for attention scores among each model using the

spatio-temporal grounding *mAP* IoU@0.4 on our GroundingYoutube dataset as shown in Table 3.3. We find 0.01 to be a reasonable threshold among all models. We will add the numbers to our supplement.



(a) Frying egg.



(b) Flip pancake.

Figure 3.10: **Visualization of spatio-temporal grounding on 3D tube.** The green tube is the GT box, and the line in the figure is the point with the max value in the attention heatmap.

3.5.7 Qualitative results

We visualize our spatio-temporal result on the GroundingYotube dataset as shown in Figure 3.9. For the GLIP model, we output the bounding box with the highest confidence score and visualize its center point. We found GLIP model focuses on the salient object while our model focuses more on

human-object interaction. We also include visualization using the tool provided by [136] [3] shown in Figure 3.10. We found that our result achieves the best performance by combining the ability to predict the action description label and the spatial localization correctly.

3.6 Summary

In this chapter, we introduce the task of multimodal spatio-temporal action grounding and a new dataset: the GroundingYoutube annotations. We propose a method that jointly learns global representations, which encodes temporal information and local representations while learning multimodal interaction between video and text. Our experiments reveal that global and local representations serve as reciprocal information for better spatio-temporal grounding in our proposed architecture. We extensively evaluate our method on various downstream tasks, including untrimmed video spatio-temporal grounding, video spatial grounding, and action step alignment. Our approach achieves state-of-the-art performance in instruction videos and generalized well with open vocabulary human object interaction datasets.

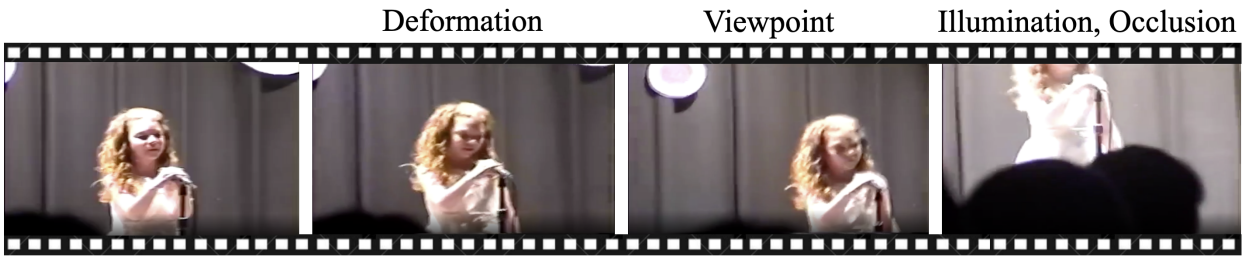
[3] https://github.com/psmmettes/spatial-aware-object-embeddings.

Chapter 4: Self-supervision from Video Tracking

4.1 Introduction

Self-supervised learning (SSL) of visual representations [137, 138, 139, 140, 5, 64, 141, 142] has become a competitive alternative to supervised learning, without requiring manually annotated labels. A key component of SSL from images is contrastive learning, a learning objective that pulls different data augmentations from the same instances (known as query and key) to be closer to each other and pushes data augmentations from different instances away. However, not all of the commonly used augmentations in images reflect the visual variability that we see in the real world. In contrast, videos provide a natural source of data augmentation, with objects undergoing deformations and occlusions, along with changes in viewpoints and illumination as shown in Figure 4.1a. As a result, recent work has tackled SSL from videos to seek more natural augmentations and meaningful semantics [143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153].

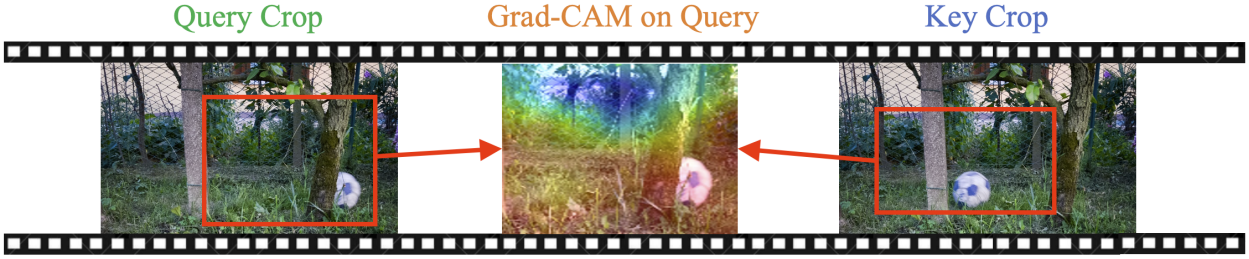
A common approach [154, 8] is to randomly sample nearby clips in videos as query and key as a natural way of data augmentation that represents the same instance since frames that are close in time are likely to share similar content. However, this sampling strategy for augmentation suffers from a few problems, as shown in Figure 4.1b and 4.1c. First, when sampling instances from a longer span of the video, the content might change substantially, resulting in samples containing totally different semantic concepts. This sampling strategy results in an imperfect supervisory signal that does not encourage semantic understanding. Second, when sampling clips from the same video, the background in the two clips are often quite similar, which allows the model to cheat by looking at the background for minimizing contrastive loss [155] as shown in Figure 4.1d. This sampling strategy leads to models learning spurious background correlations and context, which could make them less transferable and potentially biased [156].



(a) Temporal transformations in videos provide a natural source of data augmentation, making them attractive for self-supervised learning (SSL).

(b) Randomly selected query and key clips in contrastive video SSL may lead to missing objects.

(c) Query and key clips may also contain different visual concepts altogether.



(d) Since many videos contain a fixed background, SSL models can cheat by focusing on the background.

Figure 4.1: Current methods for contrastive video self-supervised learning receive an imperfect supervisory signal and can rely on background correlations when learning representations. We propose a new approach based on video tracking and Grad-CAM supervision to tackle these problems.

To alleviate these problems, we propose Pretraining with Video Tracking Supervision (PreViTS). PreViTS consists of an intelligent method to select query and key clips, which utilizes unsupervised tracking for videos. Using this freely available form of supervision, we design a temporal constraint for selecting clips that ensures that the query and the key contain the same object. In addition, using tracking information on the spatial extent of the object, we design spatial constraints to mask the background. Taken together, these spatial-temporal constraints result in better supervisory signals for contrastive learning from videos. After selecting more informative query and key clips, we train the model to learn to localize specific regions in query and key that represent the same concepts using a Grad-CAM [157]-based attention loss.

We pretrained a momentum contrastive encoder (MoCo) [5] with PreViTS on Image and Video-based SSL backbones using VGG-Sound and Kinetics-400 datasets. Evaluation on image and video downstream tasks, including image classification, object detection, instance segmentation, and action recognition, shows that PreViTS-trained models learn more accurate visual representations. In particular, training with PreViTS shows state-of-the-art performance on video action classification. Due to its ability to localize objects, PreViTS-trained models can perform unsupervised tracking across arbitrary lengths of videos, as shown by our experiments on the DAVIS challenge [158]. Additional experiments on image and video datasets with background changes show that models trained with PreViTS are less dependent on background correlations and more robust to background changes in visual classification.

In sum, our work shows that existing methods for contrastive SSL from videos do not efficiently use temporal transformations of objects. By designing a better clip sampling strategy and a loss function that encourages object localization, we are able to learn more accurate visual representations from the video that are robust to background and context changes.

4.2 Related Work

Self-supervised representation learning (SSL). Contrastive SSL approaches learn image representations [159, 160, 140, 161, 64, 76, 162] by forming positive and negative pairs, and maximizing

the similarity of positive pairs as compared to negative pairs. Positive pairs are generated from a single image instance through artificial data augmentations such as random cropping, resizing, color distortion, and Gaussian blur [64]. Going beyond learning representations from images, different frames of videos provide natural viewpoint changes and temporal information which can help learn better representations in a self-supervised manner [163, 164, 143, 146, 165, 166, 167, 168, 169]. Saliently, contrastive learning-based methods [154, 152, 153, 170, 171] that sample positive pairs from the same video have shown that view-point invariant representations can be learnt from videos. Unlike previous methods [164, 152] that sample positive pairs from unsupervised proposals with bounding boxes, we introduce an approach for sampling pairs based on spatial and temporal constraints obtained using unsupervised saliency maps, coupled with Grad-CAM supervision [157] to learn better grounded representations.

Grounded Representation Learning. Our work is also related to recent work on learning better grounded representations. Henaff *et al*. [172] introduced DetCon, a self-supervised objective which tasks representations with identifying object-level features across different image augmentations. Mo *et al*. [173] introduced a technique to mix backgrounds of different images during contrastive pretraining and showed that it leads to models learning reduced contextual and background biases. Xie *et al*. [174] propose an object-level pretraining approach for learning from complex scenes. CAST [175] learns visually grounded representations through saliency supervision. FAME [176] extracts moving foreground by frame difference and color statistics to alleviate background bias.

4.3 Method

We propose Pretraining with Video Tracking Supervision (PreViTS) to learn visual representations from videos by utilizing unsupervised object tracking. PreViTS is generalizable to both image and video models. First, we will review the standard image and video representation learning framework and then discuss our approach.

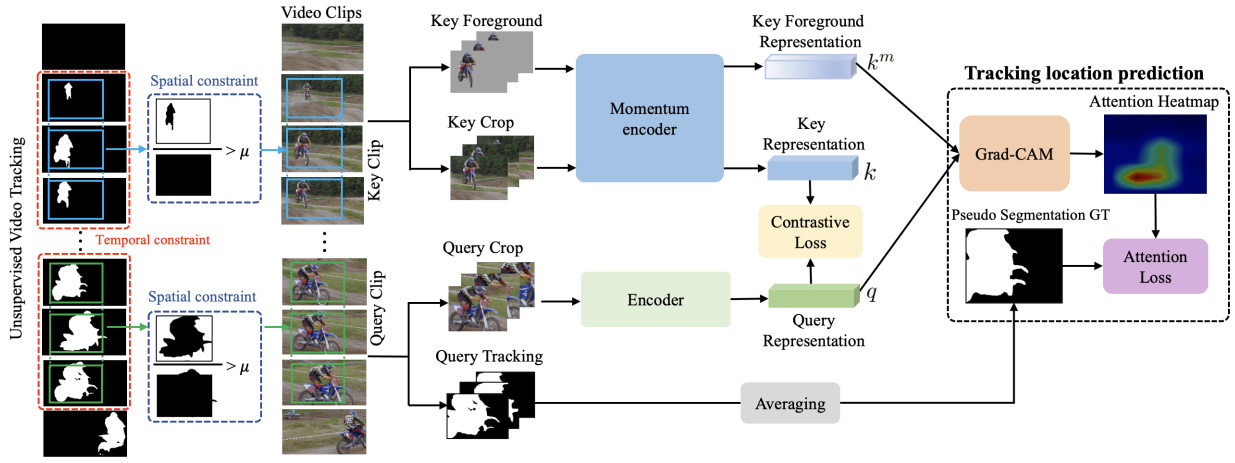


Figure 4.2: **Pretraining with Video Tracking Supervision (PreViTS):** Given an input video, we perform unsupervised tracking and apply temporal constraints to extract continuous frames that contain the tracked object region. We then apply IoU based spatial constraints to sample query and key video clips along with their masks. The encoder representations for the query and key are aligned through a contrastive loss. We then mask the key and use Grad-CAM to localize the regions in the query that maximize the (key foreground, query) similarity. We then supervise Grad-CAM with the tracked query mask using a cosine distance loss to encourage models to rely on appropriate salient object regions during contrastive pretraining.

4.3.1 Pretext-tasks in video self-supervised learning

When performing contrastive learning on videos, the positive pairs are clips from the same video selected from different times, while the negative pairs are formed with clips taken from other videos. In this work, we build our approach on top of the Momentum Contrast (MoCo) [5] model, which uses the InfoNCE [139] objective and stores the negative samples in a dynamic memory bank with a moving average encoder. Formally, given a video V, we learn feature representations for query q and key k sampled from the same video. The goal is to pull the feature distance of the positive pairs q and k to be closer and push the features of query q away from a negative set of features from other videos $N = \{n_1, n_2, ..., n_m\}$. The MoCo loss is:

$$\mathcal{L}_{\text{MoCo}} = -\log \frac{\exp(q \cdot k)/\tau)}{\sum_{n \in \{N,k\}} \exp(q \cdot n)/\tau)}, \tag{4.1}$$

where τ is the temperature constant. In our image-based model, we apply MoCo with this sampling strategy and set the length of query and key to 1 frame to extract individual frames from the same video as positive pairs.

In the video model, in addition to the MoCo contrastive loss, we also use the **relative speed prediction task** which has been found to be beneficial to understand the relative speed between the video segments proposed in **RSPNet** [8]. We sample three video segments, with two segments having the same speed and another with a different speed. The goal is to pull the feature distance for segments with the same speed closer together while pushing the features for the segment with different speed away. A triplet loss [177] is applied as follows,

$$\mathcal{L}_{Speed} = \max(0, \gamma - (pair^{+} - pair^{-})), \tag{4.2}$$

where the distance of positive pairs $pair^{+}$ should be larger than the negative pairs $pair^{-}$ by a margin $\gamma > 0$.

4.3.2 Unsupervised tracking in videos

In order to select query and key clips from the same video that contain the same visual concepts, we propose to use unsupervised object tracking to guide clip selection. To acquire unsupervised tracking information from the video we first use Deep-USPS [178], an unsupervised saliency prediction algorithm, to obtain a saliency map for the initial frame in the video. We use this saliency map as the target object for tracking and apply SORT [179], a tracking algorithm which checks the IOU constraint across continuous frame masks to track the target object through the video. Formally, given an input video V with height h, width w and temporal length t, we acquire the video object segmentation map $M \in \{0, 1\}^{h \times w \times t}$, where $M_{ijk} = 1$ indicates pixel (i, j, k) is salient, and area of salient region in time t is $A^t_M = \sum_{i,j} M_{i,j}$. The saliency map is a binary mask. Since a large majority of the web videos (and as a result, videos in vision datasets) are centered on a single object, we only utilize one (the largest) salient region in the video for tracking and do not consider multiple objects in this work.

4.3.3 Spatial-temporal cropping based on video tracking:

Once we obtain the tracking tube for the video, we constrain our random sampling to video segments covered by the tracking tube as shown in left half of Figure 4.2, where $A^t_M \neq 0$. This

ensures our sampled query and key clips will contain meaningful instances of the same object in the video. In addition to this temporal constraint, we set a spatial constraint (Figure 4.2): the random crop for the query or key should have at least $\mu \in [0, 1)$ IOU with the tracking mask. This spatial constraint tries to ensure that the query and key contain the same object for contrastive pretraining. In sum, we acquire two 3D masks for the video segment M_q and M_k, which represent the mask of the *query* and *key* containing salient regions.

4.3.4 Pretraining with Video Tracking Supervision (PreViTS)

PreViTS aims to encourage the model to learn to localize specific regions within the query and key that represent the same concept. We first determine the regions that the network relies on when matching the object regions in the key, x^k with that of the query, x^q. To obtain the object regions in key, we mask the key with the video segmentation mask, M_k, as a filter to get the key foreground, $x^{k_m} = x^k * M_k$. To understand the importance placed by the network on specific crop regions when contrastively matching their representations, similar to CAST [175], we compute Grad-CAM [157] in a contrastively-trained fashion. We do this by first forward propagating the key foreground, x^{k_m}, and the query, x^q, through the respective encoders to get k^m and q. To get the regions that would help maximizing their similarity, we take their dot-product and compute the gradients *wrt* the last convolution layer activations of the query encoder, f_q, as follows:

$$\alpha_q = \overbrace{\sum_{i,j}}^{\text{global pooling}} \underbrace{\frac{\partial q \cdot k^m}{\partial A^{f_q}_{conv5}}}_{\text{gradients via backprop}} \tag{4.3}$$

where the α_q represents the last convolutional layer neurons' importance for maximizing the similarity of the query and the key foreground representations. Through a weighted combination of α_q with the last convolutional layer activations $A^{f_q}_{conv5}$ and clipping them at zero, we can get Grad-CAM maps, G_q.

$$\mathrm{G}_q = ReLU \underbrace{\left(\sum_n \alpha_{\mathrm{q}} A^{f_{\mathrm{q}}}_{conv5} \right)}_{\text{linear combination}}. \tag{4.4}$$

Higher values in G_q represents the regions the network relies on when mapping query to key foreground.

We would ideally want the network to only rely on the tracked object regions in the query that are highlighted in the key foreground. Therefore, we apply a cosine-distance based attention loss to encourage the Grad-CAM heatmap G_q to be close to tracked object mask in the query segment M_q. This enforces the model to learn similar representations for the object irrespective of the viewpoint and transformation changes that might be present in the clips when the frames are temporally far away. The Attention loss is defined as:

$$\mathcal{L}_{\text{att}} = 1 - \frac{G_q \cdot M_{\text{q}}}{\|G_q\| \, \|M_{\text{q}}\|}. \tag{4.5}$$

Our full model is trained to minimize the sum of the losses described above. For image-only models, we apply MoCo loss and Attention loss:

$$\mathcal{L}_{\text{Image}} = \mathcal{L}_{\text{MoCo}} + \lambda \mathcal{L}_{\text{Att}}. \tag{4.6}$$

For video models, we also add the speed loss L_{Speed}.

4.4 Experiments

We aim to show that training video self-supervised models with PreViTS leads to better representations that obtain improved transfer learning performance with reduced dependence on background signal and context. We validate this by pretraining representations on two datasets and transferring them to various image and video tasks.

4.4.1 Implementation details

We pretrain our models on two datasets independently, both of which consist of 10 second-long videos at 25 FPS: (1) The training set of **VGG-Sound** [180], which contains 200k videos collected from YouTube. VGG-Sound was collected with the objective of creating an audio-visual dataset with diverse sounds and contains 300 classes as defined by audio labels. It contains a wider variety

	Method	Dataset	VOC07 clf.	IN-1k clf.	PASCAL VOC Detection			COCO Instance Segmentation					
			mAP	Top-1 acc.	AP^{bbox}_{all}	AP^{bbox}_{50}	AP^{bbox}_{75}	AP^{bbox}_{all}	AP^{bbox}_{50}	AP^{bbox}_{75}	AP^{mask}_{all}	AP^{mask}_{50}	AP^{mask}_{75}
1)	Random Init		–	–	33.8	60.2	33.1	36.7	56.7	40.0	33.7	53.8	35.9
2)	ImageNet Fully Sup		–	–	53.5	81.3	59.1	38.9	59.6	42.7	35.4	56.5	38.1
3)	MoCo	K400	69.3	47.3	50.6	78.0	55.1	40.5	58.9	41.9	35.1	55.6	37.3
4)	+ Tracking Con. Sampling	K400	70.4$_{+1.1}$	48.2$_{+0.9}$	51.2$_{+0.6}$	78.4$_{+0.4}$	56.1$_{+1.0}$	40.8$_{+0.3}$	59.5$_{+0.6}$	42.6$_{+0.7}$	35.8$_{+0.7}$	56.8$_{+1.2}$	38.3$_{+1.0}$
5)	+ PreViTS	K400	71.2$_{+1.9}$	48.6$_{+1.3}$	51.8$_{+1.2}$	78.3$_{+0.3}$	56.0$_{+0.9}$	41.0$_{+0.5}$	59.4$_{+0.5}$	42.8$_{+0.9}$	35.6$_{+0.5}$	57.2$_{+1.6}$	38.4$_{+1.1}$
6)	MoCo	VGG Sound	68.3	46.9	48.3	76.5	52.6	38.4	58.7	41.9	35.0	55.8	37.2
7)	+ Tracking Con. Sampling	VGG Sound	70.3$_{+2}$	48.1$_{+1.2}$	49.0$_{+0.7}$	77.1$_{+0.6}$	52.7$_{+0.1}$	38.3$_{-0.1}$	58.7$_{+0.0}$	41.7$_{-0.2}$	35.0$_{+0.0}$	55.9$_{+0.1}$	37.6$_{+0.4}$
8)	+ PreViTS	VGG Sound	73.0$_{+4.7}$	50.6$_{+3.7}$	52.5$_{+4.2}$	78.7$_{+2.2}$	55.1$_{+2.5}$	39.4$_{+1.0}$	59.8$_{+1.1}$	43.0$_{+1.1}$	35.7$_{+0.7}$	56.8$_{+1.0}$	38.2$_{+1.0}$

Table 4.1: **Transfer Learning on Image Downstream Tasks:** On tasks using linear probes (VOC and ImageNet classification) and finetuning (VOC Detection, COCO Segmentation), PreViTS outperforms baseline MoCo when evaluated on models pretrained on VGG-Sound and Kinetics-400. We color the difference ≥ 0.5 to show improvement over the baseline MoCo models (row 3 and 6).

of object classes and higher object-centricity as compared to action classification datasets common in the video understanding literature. (2) The **Kinetics-400** dataset [10], which consists of around 240k training videos with 400 human action classes. Kinetics-400 is a widely-used dataset, which enables us to compare PreViTS's performance to prior methods.

For experiments with the image model, we use the ResNet-50 backbone and sample one frame with 224×224 spatial sizes for each clip. For experiments with the video model, we use an S3D-g backbone and sample 16 continuous frames with 224×224 spatial sizes for each clip. We perform standard data augmentation on clips, including random Gaussian blur, and random color jitter [64]. All models are trained with 200 epochs with SGD and a batch size of 256. We apply a cosine learning rate scheduler with an LR of 0.03 for the image model and 0.5 for the video model. Following He *et al*. [5], we set $\tau = 0.07$, $K = 65535$, $\gamma = 0.15$, $\mu = 0.3$, $\lambda = 3$. We train PreViTS with 16 A100 GPUs. The training time is two days for pretraining VGG-Sound and three days for pretraining on Kinetics. We will release the code for replicating our work upon publication. For both image and video tasks, we compare with the following baselines: (1) **Random Init** of weights without pretraining, (2) **MoCo/RSPNet** to demonstrate standard self-supervised model performance for image (MoCo) and video (RSPNet), (3) **MoCo/RSPNet + Tracking Constrained Sampling** to evaluate our unsupervised tracking-based spatial-temporal sampling strategy.

Image model is from MoCo, video model is from RSPNet. For experiments with the image model, we use the ResNet-50 backbone and sample one frame with 224×224 spatial sizes for

each clip. For experiments with the video model, we use an S3D-g [128] backbone and sample 16 continuous frames with 224 $\times$ 224 spatial sizes for each clip. We perform standard data augmentation on clips, including random Gaussian blur, and random color jitter [64]. To compare with other baseline methods, we also trained on R(2+1)D[20], and C3D[18] backbone following [8]. We followed [8] to train our model with 200 epochs with SGD and a batch size of 256. We apply a cosine learning rate scheduler with an LR of 0.03 for the image model and 0.5 for the video model. Following He *et al*. [5], we set $\tau = 0.07$, $K = 65535$, $\gamma = 0.15$, $\mu = 0.3$, $\lambda = 3$. The training time is two days for pretraining VGG-Sound and three days for pretraining on Kinetics. For both image and video tasks, we compare with the following baselines: (1) **Random Init** of weights without pretraining, (2) **MoCo/RSPNet** to demonstrate standard self-supervised model performance for image (MoCo) and video (RSPNet), (3) **MoCo/RSPNet + Tracking Constrained Sampling** to evaluate our unsupervised tracking-based spatial-temporal sampling strategy.

4.4.2 Image recognition tasks

We evaluate our learned features on four downstream image recognition tasks: **(a)** PASCAL VOC [181] linear classification, **(b)** ImageNet-1k [72, 83] linear classification, **(c)** PASCAL VOC object detection, and **(d)** COCO [182] instance segmentation. Following [183, 175], for **(a, b)**, we perform linear classification by using the SSL model as a frozen feature extractor and training a classifier on top. For **(c, d)**, we use the SSL model as weight initialization for fine-tuning on the labeled datasets. Detailed experimental settings can be found in the supplementary.

Our results in Table 4.1 show that training PreViTS outperforms baseline MoCo training on all tasks, obtaining robust gains in VOC and ImageNet classification, along with VOC detection and COCO instance classification. Notably, the performance gains when pretraining on VGG-Sound are larger as compared to those on Kinetics-400, even though Kinetics-400 is 20% larger in terms of the number of videos. We speculate that due to VGG-Sound containing a more diverse collection of objects as compared to Kinetics-400, which is primarily human action-centric, VGG-Sound benefits more from being able to learn object-focused representations when training with PreViTS. The

performance improvement over baseline is especially large on the VOC detection task, aided by the improved ability to localize objects during pretraining. Finally, while it is typically challenging to obtain comparable performance to supervised ImageNet pretraining using video SSL pretraining on image recognition tasks [152], due to the larger domain shift, MoCo models trained with PreViTS still obtain comparable or better performance to ImageNet-fully supervised training on VOC detection and COCO instance segmentation tasks.

Method	**Dataset**	**UCF-101**
RSPNet	VGG Sound	86.4
+ Tracking Constrained Sampling	VGG Sound	$87.5_{+1.1}$
+ PreViTS	VGG Sound	$88.9_{+2.5}$
RSPNet	K400	87.6
+ Tracking Constrained Sampling	K400	$89.1_{+1.5}$
+ PreViTS	K400	$91.8_{+4.2}$

Table 4.2: **Video Action Classification:** Training with PreViTS obtains significant performance gains on the commonly-evaluated downstream task of UCF-101 action recognition.

4.4.3 Video tasks: Action recognition

To evaluate the performance of PreViTS-trained models on video classification tasks, we perform action recognition on the UCF-101 dataset [33]. Following Xu *et al.* [191], in all experiments, we finetune our pretrained model on labeled videos with 50 epochs using a learning rate of 0.05. We drop the projection head and replace it with a randomly initialized fully-connected layer. We report top-1 accuracy on the UCF-101 dataset when pretraining with PreViTS on VGG-Sound and Kinetics-400 datasets (Table 4.2). Training with PreViTS obtains a substantial improvement over RSPNet on both pretraining datasets. Notably, the model pretrained on Kinetics-400 had better performance with RSPNet and a larger absolute improvement with RSPNet + PreViTS (4.2% versus 2.5%), over VGG-Sound. We speculate that since human actions are better represented in Kinetics-400, the representation learnt using these videos transfers better to UCF-101, and also benefits more from training with PreViTS. Finally, we compare the performance of RSPNet + PreViTS pretrained with Kinetics-400 with other state-of-the-art video SSL methods [8] in Table

Method	Input size	Params	Backbone	UCF-101
RSPNet [8]	112 × 112	33.4M	C3D	76.7
CACL [184]	112 × 112	33.4M	C3D	77.5
PreViTS	112 × 112	33.4M	C3D	**78.7**
Pace [185]	112 × 112	14.4M	R(2+1)D	77.1
STS [186]	112 × 112	14.4M	R(2+1)D	77.8
VideoMoCo [187]	112 × 112	14.4M	R(2+1)D	78.7
RSPNet [8]	112 × 112	14.4M	R(2+1)D	81.1
PreViTS	112 × 112	14.4M	R(2+1)D	**81.9**
SpeedNet [162]	224 × 224	9.6M	S3D-g	81.1
CoCLR [188]	224 × 224	9.6M	S3D-g	87.9
STS [186]	224 × 224	9.6M	S3D-g	89.0
RSPNet [8]	224 × 224	9.6M	S3D-g	89.6
ASCNet [189]	224 × 224	9.6M	S3D-g	90.8
PreViTS	224 × 224	9.6M	S3D-g	**91.8**

Table 4.3: **Comparison to prior work on UCF-101 performance:** Our best-model trained with PreViTS outperforms all existing methods for video self-supervised learning on UCF-101 downstream performance, when using comparable training resources.

4.3. With the same architecture, computational budget, epoch, batch size, and pretraining data for a fair comparison, our approach outperforms prior work and obtains state-of-the-art performance.

4.4.4 Video tasks: Video Retrieval

We also evaluate our video retrieval task on the UCF-101 dataset. Given a video as a query, we search the most relevant video by cosine distance using the nearest neighbor search. Following [8], we evaluate our method on the split 1 of UCF101 dataset and apply the top-k accuracies (k=1, 5, 10, 20, 50) as evaluation metrics. As shown in Table 4.4, our model outperforms the other baselines by a large margin, showing the effectiveness of the proposed training process.

4.4.5 Backgrounds challenge

We expect feature representations obtained using PreViTS to be less dependent on object backgrounds and context. To quantify this, we utilize the "backgrounds challenge" [192] on both image and video classification tasks as shown in Table 4.5.

Method	Top-k				
	$k = 1$	$k = 5$	$k = 10$	$k = 20$	$k = 50$
Pace [185]	31.9	49.7	59.2	68.9	80.2
RSPNet [8]	36.0	56.7	66.5	76.3	87.7
STS [186]	39.1	59.2	68.8	77.6	86.4
CACL [184]	43.2	61.1	69.9	78.2	88.2
TCLR [190]	48.6	67.6	75.5	82.5	-
PreViTS	**53.4**	**69.4**	**77.8**	**85.5**	**93.0**

Table 4.4: **Video retrieval** results on UCF101. Our model outperforms other baselines using the same architecture C3D backbone.

MoCo-VGG Sound	**Image Backgrounds Challenge [192]**							
	Original	Mixed-Same	Mixed-Rand	Mixed-Next	Only-FG	No-FG	Only-BG-B	Only-BG-T
Default	77.9	53.3	37.8	33.8	40.9	24.6	9.7	13.5
+ PreViTS	81.0$_{+3.1}$	56.9$_{+3.6}$	42.0$_{+4.2}$	38.0$_{+4.2}$	53.0$_{+12.1}$	28.0$_{+3.4}$	8.8$_{-0.9}$	13.0$_{-0.5}$
RSPNet-K400	**Video Backgrounds Challenge**							
Default	70.7	40.7	30.3	29.5	20.9	49.1	35.2	28.6
+ PreViTS	74.0$_{+3.3}$	48.0$_{+7.3}$	35.9 $_{+5.6}$	32.7$_{+3.2}$	27.8$_{+6.9}$	51.9$_{+2.8}$	33.7$_{-1.5}$	28.3$_{-0.3}$

Table 4.5: **Robustness to background changes**. On image and video Backgrounds Challenge datasets, PreViTS outperforms baselines where the foreground was included (columns 1-5), especially the Only-FG setting. Also, PreViTS-trained models are less accurate when foreground information is entirely eliminated (columns 7, 8), showing their reduced reliance on background information.

Image Backgrounds Challenge. First, we evaluate our model on the original Backgrounds Challenge [192], which was designed to test a model's robustness to various background changes. It contains 9 ImageNet classes with 450 images for each class. We evaluate our model along with the baseline model pretrained on VGG-Sound and train a linear layer with ImageNet-1K. Results show that pretraining with PreViTS achieves significant improvement on all tasks defined in the Backgrounds Challenge. Examples of different settings can be found in Figure 4.3. In the Only-FG setting, where the background is set to black, PreViTS obtains an absolute improvement of 12.1%, showing that it is less dependent on background information. When backgrounds are swapped (Mixed-Same, Mixed-Rand, Mixed-Next), PreViTS obtains an absolute improvement of 3.6 – 4.2%, indicating that representations learnt with PreViTS reduce the reliance on background correlations. There is a slight increase in performance in the No-FG setting, likely due to the model learning contour information from videos. However, in settings where no information from the foreground

Figure 4.3: **Video Background Challenge:** We evaluate PreViTS by introducing a Video Backgrounds Challenge to evaluate background-robustness of video models. FG = foreground, BG = background. Foreground-background combinations include: Only-BG-B (FG: Black, BG: Unmodified), Only-BG-T (FG: Tiled background, BG: Unmodified), Mixed-Same (FG: Unmodified, BG: Random BG of the same class), Mixed-Rand (FG: Unmodified, BG: Random BG of a random class), and Mixed-Next (FG: Unmodified, BG: Random BG of the next class.)

is provided (Only-BG-B, and Only-BG-T), PreViTS obtains lower accuracy than baseline, which reinforces that it is less dependent on the background signal.

Video Backgrounds Challenge (JHMDB). Taking inspiration from the image Backgrounds Challenge, we construct a new Video Backgrounds Challenge to test background-robustness on videos. We use the JHMDB dataset [193]—consisting of 21 HMDB [34] action recognition classes with 50 videos per class—for which the ground truth foreground mask is available. We follow Xiao et al. [192] to construct 8 foreground-background combinations (Figure 4.3) for JHMBD. We evaluate performance using a model trained on Kinetics-400 and finetuned on UCF-101. Models trained with PreViTS outperform the baseline model (RSPNet) in all settings. Similar to the trends on Image Backgrounds Challenge, PreViTS obtains significant improvement in settings where the background is set to black or is replaced by background from another video. In settings where the foreground is removed, we find the accuracy drop to be higher for PreViTS compared to baseline (22.1 vs. 21.6). Video representation learning models have been shown to suffer from over-reliance

on background information, called representation bias [194] or scene bias [156]. Training with PreViTS can help mitigate this bias.

Video Backgrounds Challenge (mini-Kinetics). In addition to the video backgrounds challenge, we also evaluate robustness to background signal on the mini-Kinetics dataset [195], a subset of Kinetics-400 designed to study if video classification models depend on the background signal for scene classification. This dataset contains foreground bounding boxes computed by a person detection model. We utilize the bounding boxes to mask the foreground object to analyze if the model depended on scene features when performing action classification. The model with PreViTS achieved an accuracy of 55.24% in the Original setting compared to 47.18% for the baseline RSPNet. When the foreground was masked (No-FG), the accuracy for PreViTS drops by 6.9%, as compared to a drop of 2.71% for the baseline model, indicating that the PreViTS-trained model relies less on the background signal.

Method	Occlusion		Viewpoint		Illumination Dir.		Illumination Color		Instance		Instance+Viewpoint	
	Top-10	Top-25	Top-10	Top-25	Top-10	Top-25	Top-10	Top-25	Top-10	Top-25	Top-10	Top-25
MOCO	83.25	76.45	84.83	75.31	85.09	74.74	99.42	95.88	48.99	43.55	51.23	46.83
Region Tracker [152]	83.26	76.52	84.97	76.18	88.30	79.34	99.77	97.70	48.81	44.38	53.31	49.04
PreViTS	**85.11**	**78.84**	**89.35**	**81.28**	**91.66**	**83.94**	**99.92**	**98.89**	**55.45**	**49.09**	**56.97**	**51.70**

Table 4.6: **Invariances of Video representations:** The representation learned by PreViTS is more invariant to various transformations as compared to baseline MoCo, as shown by the top-k Representation Invariance Score (RIS) [152]. The large improvement in viewpoint invariance is likely due to our strategy of sampling tracked objects with different viewpoints. The large improvement in instance invariance shows that PreViTS is better at learning object concepts instead of low-level pixel similarities. Improved invariance is useful for object recognition tasks. See Section 4.4 for details of RIS.

4.4.6 Invariances captured by PreViTS.

We expect representations learnt by PreViTS to have better invariance to various transformations (occlusion, viewpoint, illumination, instance), due to more effective use of object instance information during contrastive learning. Following [152], we measured the representation's invariances when predicting classes using the top-k Representation Invariance Score (RIS). We selected top-10/25 neurons from encoder with similar activation behavior between transformations and computed

its mean score. PreViTS is significantly more invariant to transformations than other baselines (Table 4.6).

Region Similarity $\mathcal{J}$	Mean $\mathcal{M} \uparrow$	Recall $\mathcal{O} \uparrow$	Decay $\mathcal{D} \downarrow$
MoCo	0.315	0.638	0.025
+ PreViTS	0.544	0.769	-0.014

Table 4.7: **Unsupervised Tracking on DAVIS 2016:** We show that through our grounding supervision, we are able to better track objects across videos of arbitrary lengths given just the first frame and its associated segmentation map.

4.4.7 Video tracking evaluation

To demonstrate grounding and tracking ability, we evaluate our model on the single object video tracking dataset [158] in Grad-CAM attention fashion. In the original video tracking task, the input is the first frame of the video along with the foreground segmentation mask. The goal is to predict the pixel-level mask of the foreground in the later video frames. In our setting, we utilize our pipeline as shown in Figure 4.2 to perform tracking. We feed the first frame and its segmentation to acquire the key foreground. Then, we feed the later frames as queries and compute the Grad-CAM attention heatmap to localize the corresponding region in the later frames. Since the attention heatmap resolution is 7×7, we cannot perform pixel-level prediction. Our evaluation metrics follow [158] and compute: Region similarity ($\mathcal{J}$), which represents the IOU between the predicted foreground mask and GT foreground mask; Mean ($\mathcal{M}$) is the average value of $\mathcal{J}$; Recall ($\mathcal{O}$) evaluates the fraction of sequences scoring higher than a threshold; Decay ($\mathcal{D}$) evaluates the averaged performance drop over time, e.g., $\mathcal{J}_{t=4} - \mathcal{J}_{t=1}$. As shown in Table 4.7, PreViTS outperforms the baseline MoCo by a significant margin, which demonstrates our model's ability to localize objects in dynamic videos. Figure 4.4 shows how PreViTS is able to localize objects while the baseline fails when the object appears in a novel viewpoint (Figure 4.4(d)).

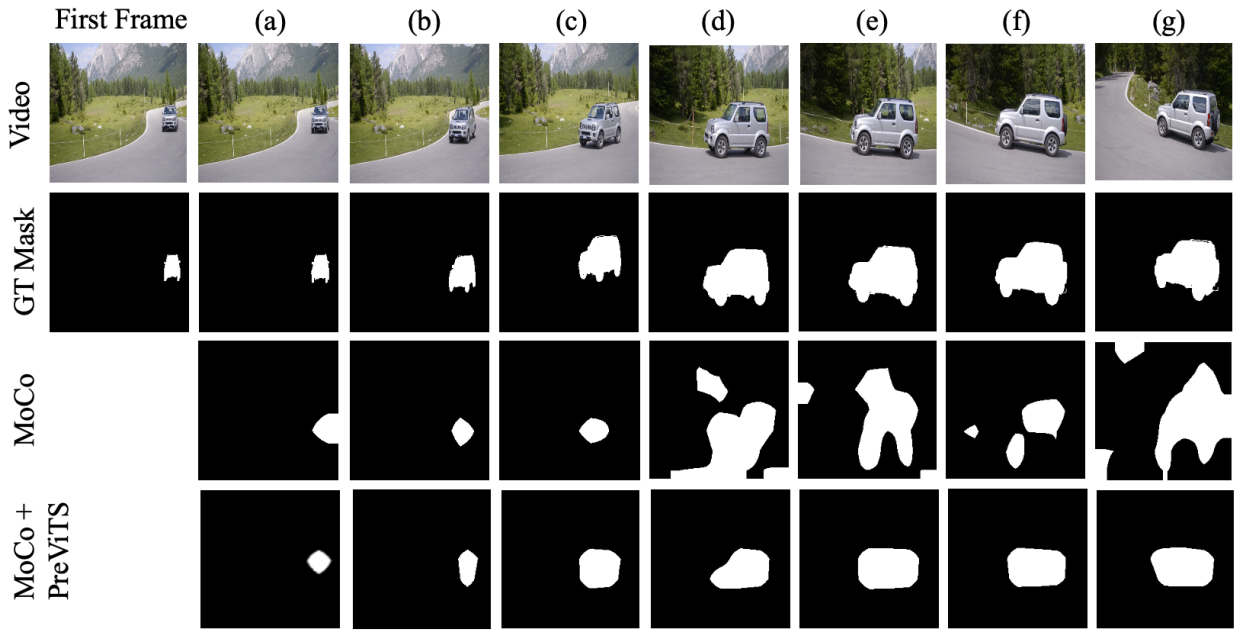


Figure 4.4: **Unsupervised Object tracking**. Using Grad-CAM attention and the query-key framework, PreViTS-trained model can be used to track objects across the video given the first frame and corresponding segmentation map of the object to track. PreViTS is able to localize objects under viewpoint changes, while the baseline model is unable to do so.

4.4.8 Ablation Studies

Next, we conduct an ablation study on the effect of different design decisions on performance. We evaluate the image model trained on the VGG-Sound dataset for 200 epochs and evaluate the video model trained on the Kinetics-400 dataset for 50 epochs following [9].

Temporal distance constraint between positive pairs: First, we investigate the effect of different temporal sampling strategies in Table 4.8a. We define δ to be the temporal distance between the query and key segment. $\delta = 0$ uses the same sample segment for query and key with image augmentation. **Constant** δ samples query and key in a fixed length of 1.7 sec, which ends up as an easier task for the model to learn and does not generalize to the downstream task. **Varying** δ does not constrain the distance between the clips. We find this setting to perform the best as it enables the network to localize regions across the clips irrespective of their temporal distance.

Effect of area threshold μ: We apply spatial constraint when sampling our positive pairs where the crop should cover at least μ IOU of the tracking object area. Here, we investigate the different

Temporal Sampling	Varying δ	Constant δ	$\delta = 0$
VOC07	73.0	72.4 $_{-0.6}$	67.5 $_{-5.5}$
UCF-101	84.5	83.7 $_{-1.8}$	84.3 $_{-0.2}$

(a) Effect of different temporal sampling strategy.

Spatial area threshold	$\mu = 0.0$	$\mu = 0.2$	$\mu = 0.3$	$\mu = 0.4$
VOC07	71.5 $_{-1.5}$	72.1 $_{-0.9}$	73.0	72.8 $_{-0.2}$
UCF-101	83.7 $_{-3.7}$	85.1 $_{+0.6}$	84.5	84.2 $_{-0.3}$

(b) Effect of Area threshold μ (Fixing $\mu = 0.3$)

Loss weighing factor	$\lambda = 0.0$	$\lambda = 2.0$	$\lambda = 3.0$	$\lambda = 4.0$
VOC07	70.3 $_{-2.7}$	72.4 $_{-0.6}$	73.0	72.6 $_{-0.4}$
UCF-101	80.8 $_{-3.7}$	83.4 $_{-2.1}$	84.5	84.1 $_{-0.6}$

(c) Effect of loss weighing factor λ (Fixing $\lambda = 3.0$)

Tracking supervision	No Tracking	Unsupervised	Supervised
VOC07	68.3 $_{-4.7}$	73.0	75.0 $_{+2.0}$
UCF-101	79.0 $_{-5.5}$	84.5	86.1 $_{+1.6}$

(d) Effect of different tracking supervision

Table 4.8: **Ablations for PreViTS training:** We isolate the effects of our training components. We find that **(a)** randomly sampling without temporal distance constraints leads to the best performance, **(b)** adding some amount of spatial constraints based on IoU with tracking mask ensures that different clips contain common salient regions and this improves performance, **(c)** increasing weights on attention loss increases the downstream performance up to a certain point, **(d)** replacing unsupervised video tracking supervision with supervised tracking improves downstream performance slightly.

values of μ in the range 0 to 0.4. Results in Table 4.8b demonstrate that adding spatial constraint helps the model focus on meaningful objects in the video, but enforcing a very strict constraint hurts the performance as it limits the variation while sampling. We find $\mu = 0.3$ to be optimal.

Effect of loss weight λ: We test different loss weights λ to balance between the two losses. Results from Table 4.8c show that non-zero values of λ outperform $\lambda = 0.0$, indicating that attention loss is important in PreViTS. Higher λ improves performance up to a point—performance improves with λ= 2.0, 3.0, and slightly degrades with λ= 4.0. We find $\lambda = 3.0$ to be optimal.

Supervised v.s. unsupervised tracking supervision: To understand the effect of the quality of tracking supervision, we evaluate our model using the tracking information provided by a supervised model [196] in Table 4.8d. The model trained with this supervision has better downstream

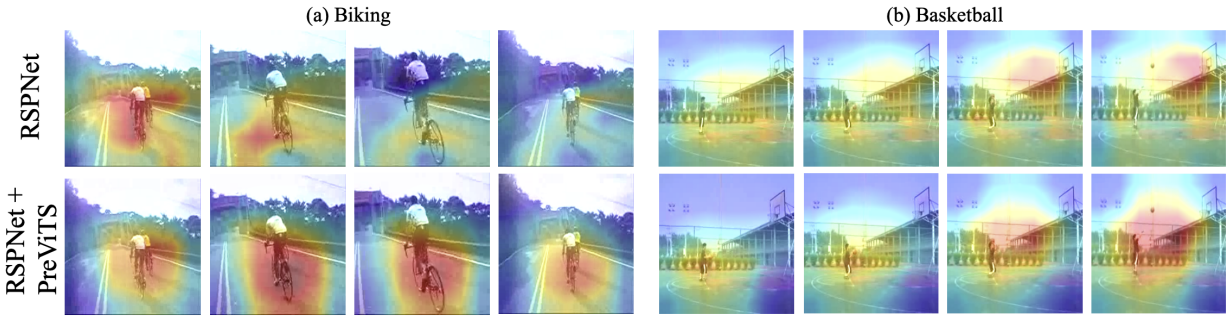


Figure 4.5: **Visual Grounding for Action Classification.** PreViTS provides better visual grounding as shown by Grad-CAM attention maps of pretrained models finetuned on UCF-101. In (a), our model focuses on the human and bike while the baseline model attends to seemingly irrelevant regions, including the road in the background. In (b), our model attends to the man and the ball in the air in addition to the basketball court while the baseline model focuses mostly on the court.

performance, but the performance drop by switching to unsupervised tracking is still acceptable.

4.4.9 Visual grounding and localization

Finally, we visualize the grounding and localization ability of PreViTS-trained models finetuned on UCF-101 using Grad-CAM. Our model has a better grounding ability as compared to the baseline and focuses on foreground objects instead of background scenes (Figure 4.5). In Figure 4.6, we provide a query with two different segmentation corresponding to the different foreground objects. We feed the query and the key foreground into the PreViTS-trained model to compute the Grad-CAM attention heatmaps. Given the different key foreground, our model can localize the man and ball, respectively. At the same time, the attention heat map in the baseline is more spread out and cannot generate discriminative attention of the two objects. Note that even though PreViTS hasn't seen multi-object masks during pretraining, it is still able to localize multiple concepts discriminatively. More visualizations for UCF-101 action recognition in Figure 4.7, Video Backgrounds Challenge in Figure 4.8, and DAVIS video object segmentation in Figure 4.9 and 4.10.

4.5 Limitations and potential impact:

Our method has a few limitations. First, acquiring and utilizing unsupervised tracking requires additional computational resources. Also, since our current tracking method captures the most

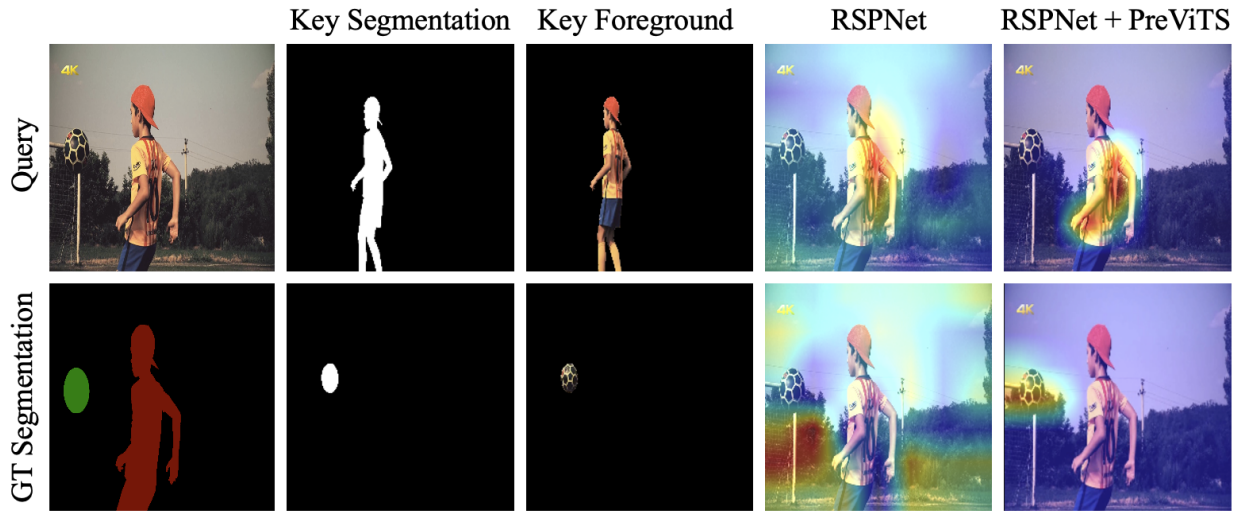


Figure 4.6: **Discriminative localization of objects.** When provided query with two different segmentation corresponding to different foreground objects and the key foregrounds, PreViTS-trained model is able to localize them accurately, capturing class-specific semantic discrimination between objects.

salient object in the video, we do not model multi-object interaction in the video, which is an interesting future work direction. Moreover, our pretraining datasets are relatively cleaner than random videos on YouTube. It is unknown if our method can generalize to the different genres such as news and gaming. Finally, our pretraining datasets may contain unintended societal, gender, racial, and other biases, whose effect was not examined in the current work.

4.6 Summary

In this chapter, we propose a novel visual self-supervised network that learns to localize foreground objects present in video data utilizing unsupervised tracking supervision. Experiments on various image and video downstream show that guiding the model to focus on the foreground region is beneficial for accurate video representations self-supervised learning. Also, we demonstrate different properties of our learned features, which capture viewpoint, occlusion, and deformation invariances and have a better grounding ability. We hope our approach can enable robust and accurate visual representation learning from large-scale uncurated video data from the internet.

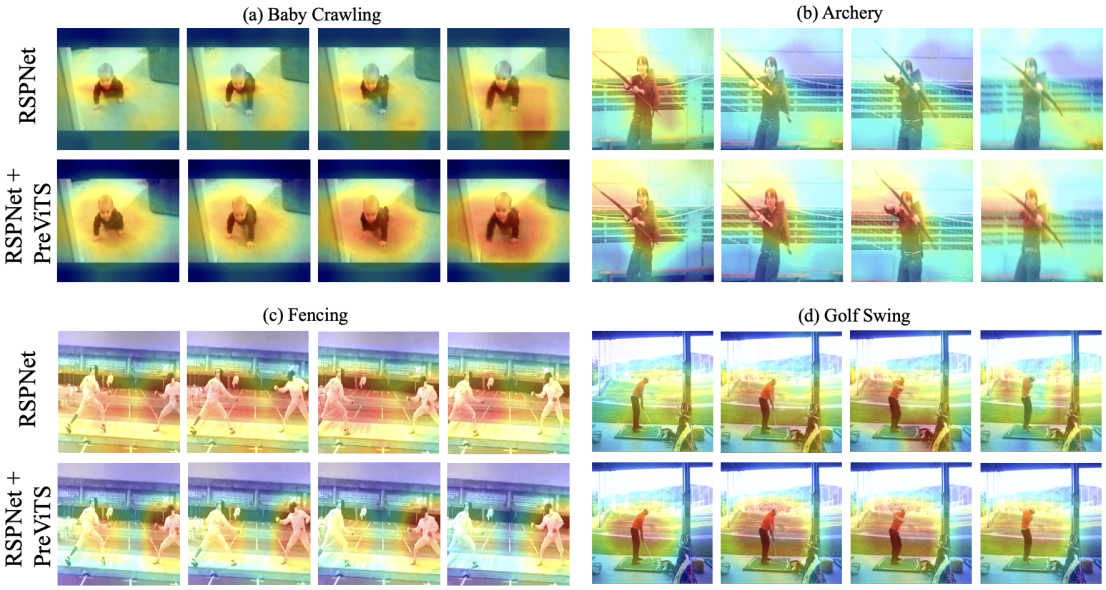


Figure 4.7: Grad-CAM Visualization for UCF-101 Action Classification.

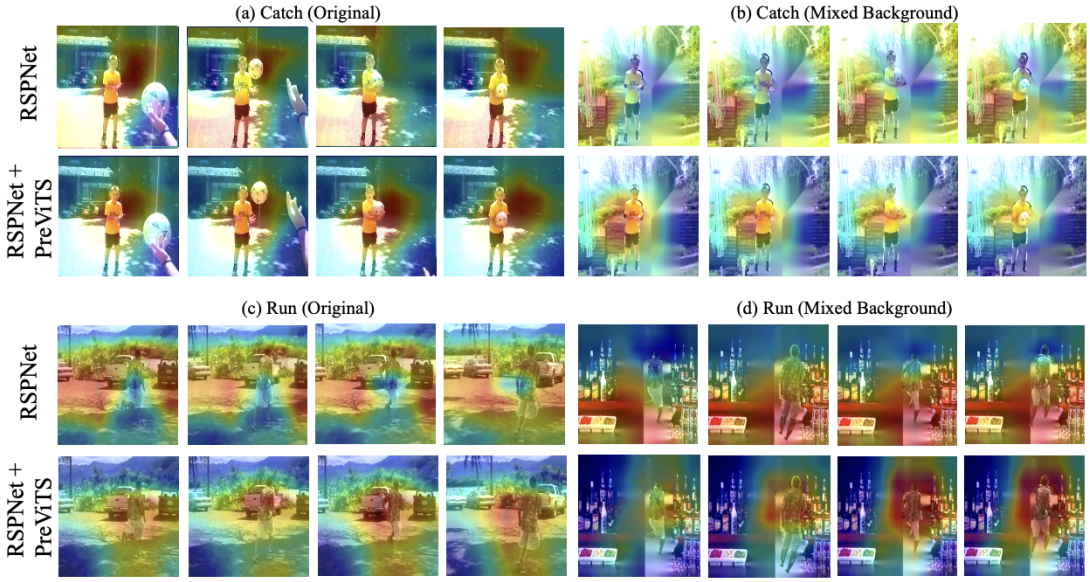


Figure 4.8: Grad-CAM Visualization for Video Backgrounds Challenge.

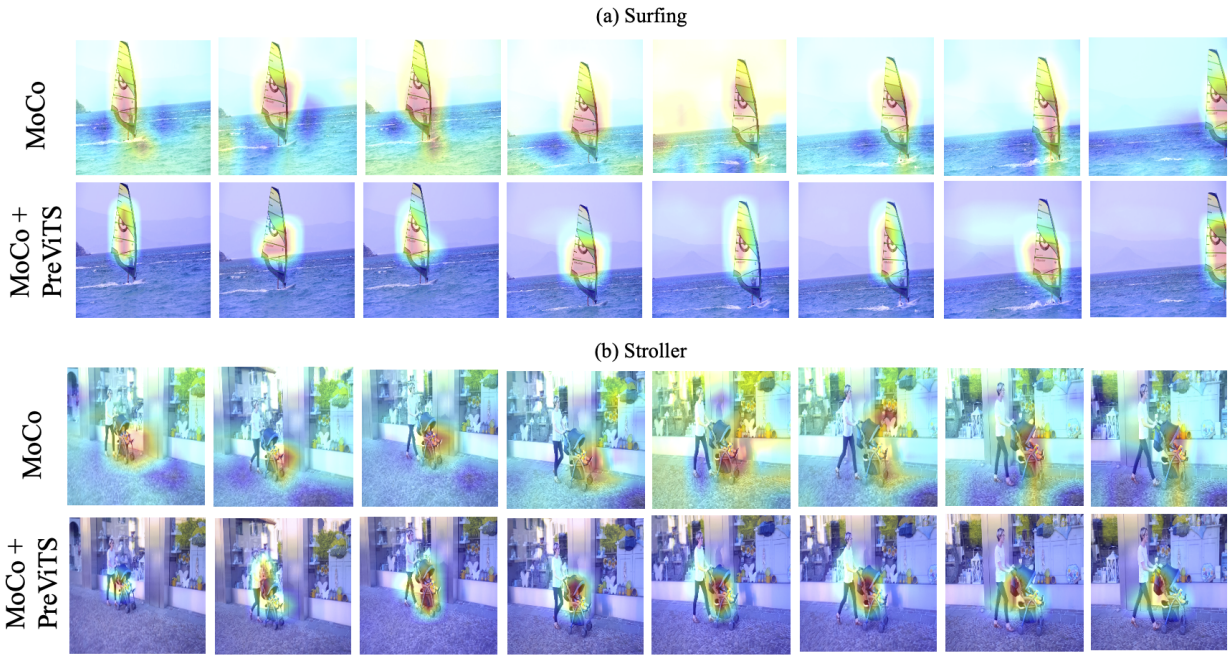


Figure 4.9: Grad-CAM Visualization for DAVIS Video Object Tracking and Segmentation.

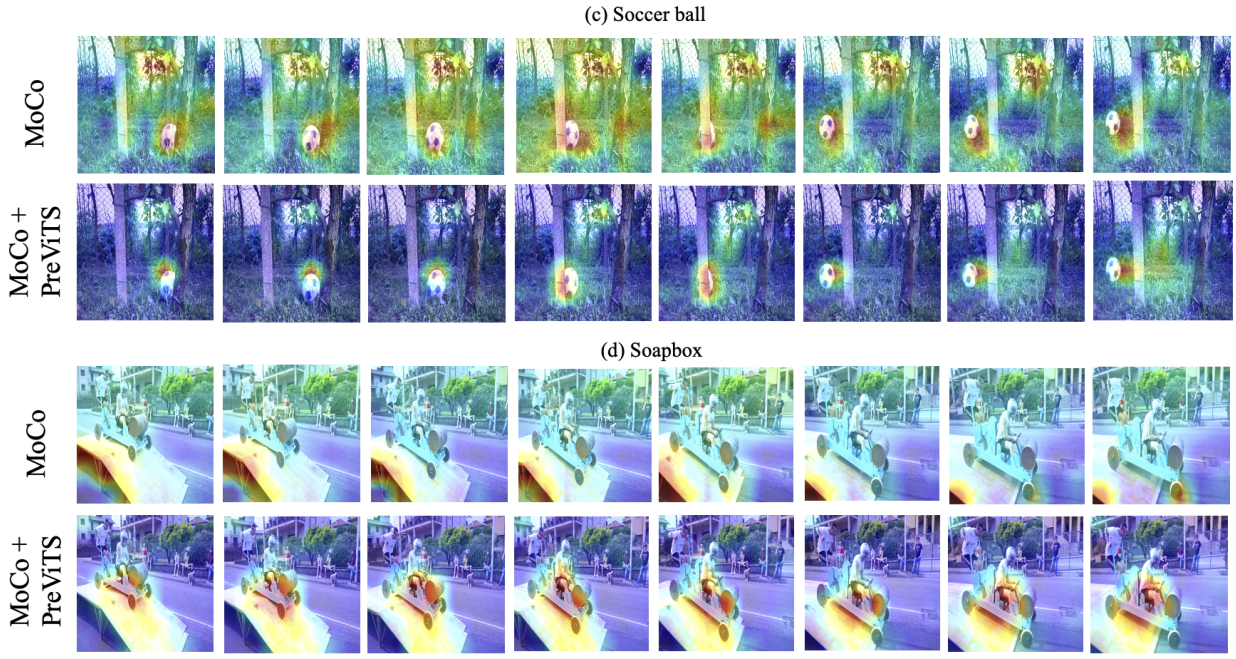


Figure 4.10: Grad-CAM Visualization for DAVIS Video Object Tracking and Segmentation.

Chapter 5: Multimodal Event Extraction

5.1 Introduction

Traditional event extraction methods target a single modality, such as text, images, or videos. However, real-world multimedia (e.g. online news) features content in multiple modalities which collectively convey a cross-modal narrative. As a consequence, components of events described by the document may lie jointly or solely in either the textual or visual modalities. By randomly watching 100 videos and associated articles from BBC Official YouTube Channel, we find that 45% of videos contain event arguments that are not explicitly mentioned in the article.

Event extraction is a well-studied problem in the natural language processing community [197, 198, 199, 200]. Similarly, methods focusing on event argument extraction have likewise been proposed [201, 202]. However, all of these methods solely target the text modality and ignore the contribution of visual media. A related line of research has emerged in the computer vision community focusing on the extraction of purely visual events [203, 204, 205, 206]. While a few methods have sought to transfer visual knowledge from images to improve text-only event extraction [207, 208], these do not detect multimodal events, whose arguments span multiple modalities.

[209] propose a method for extracting multimodal events from text and images jointly. However, [209]'s method does not handle videos. Extending [209] to the video domain is non-trivial because localizing events in videos requires first identifying temporal boundaries of the event, which is a challenging vision problem in its own right [210, 211, 212]. Moreover, while [209] transfer existing image and text event extraction resources to the multimodal domain, there are no datasets containing event argument localization in videos, thus [209]'s method cannot be directly trained for multimodal text and video event extraction as it was for images.

We argue that multimodal event extraction from videos is important for several reasons. For

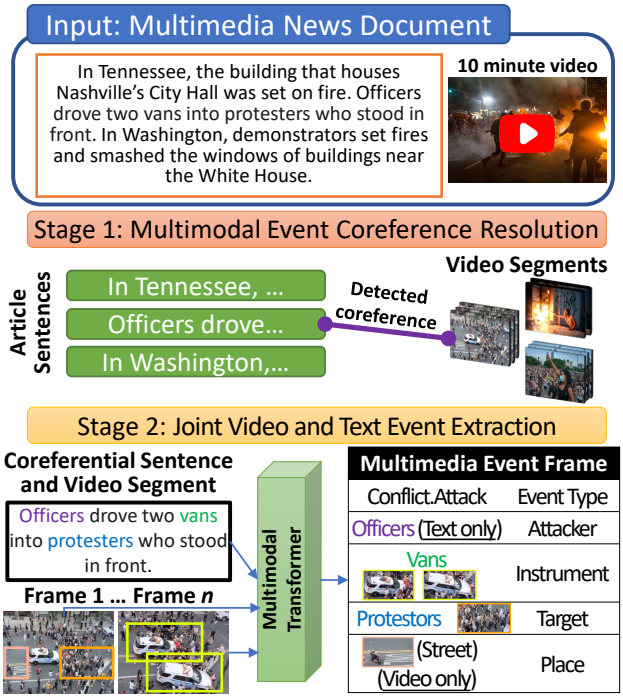


Figure 5.1: We introduce the problem of video multimedia event extraction. Given a multimedia document containing a text article and a video, the goal is to jointly extract events and arguments. Our method first performs multimodal event coreference resolution to identify which sentences and video segments refer to the same event. Our novel multimodal transformer then extracts multimedia event frames from coreferential sentence and video segment pairs. Our method is able to resolve coreference and extract multimodal event frames more accurately than existing approaches.

one thing, images contain snapshots of events, but may not capture all arguments or participants of the event in a single snapshot. In contrast, videos often contain more action events and may reveal additional event arguments that can be extracted as events evolve over time that may be missing from any single frame. Finally, we find some event argument *roles* are hard to determine from single images, while video provides additional context which helps disambiguate the roles different arguments play in the event.

In this paper, we propose the first model that extracts multimodal events and arguments from text and videos jointly. Specifically, we propose a new task called Video M^2E^2 (Video **M**ulti**M**edia **E**vent **E**xtraction). Given a document with an accompanying video, our goal is to jointly extract the events and argument roles appearing in both data modalities. Because of the lack of an existing dataset for this task, we introduce a new multimodal video-text dataset with extensive annotations covering

event and argument role extraction, coreference resolution, and grounding of event arguments (bounding boxes).

We tackle this task in a two-stage manner: first we find a coreferential sentence-segment pair and then we jointly extract events from it. For multimodal event coreference resolution, we propose a self-supervised model to find video segment-sentence pairs describing the same event. These coreferential cross-modal pairs are then used to perform event classification and argument role labeling. To do so, we propose a novel multimodal transformer architecture which learns to perform event and argument role prediction jointly from video and text. We show that this system substantially outperforms unimodal approaches, while allowing us to discover event arguments lying solely in one modality.

To summarize, we make the following contributions. We propose the novel problem of video multimodal event extraction and contribute a high-quality benchmark dataset for this task containing extensive annotations of event types, event arguments and roles, argument grounding, and cross-modal coreference resolution of events in text and videos. We propose a self-supervised training strategy which allows us to find coreferential sentence and video segment. We introduce a novel multimodal transformer architecture leveraging modality-specific decoders for joint text and video event and argument extraction. We present extensive experimental results demonstrating that our proposed approach significantly outperforms both unimodal and multimodal baselines for event coreference resolution, event extraction, and argument role labeling.

5.2 Related Work

Learning multimodal common space. Instead of learning representations in single modalities (text, visual), there have been various works that tried to learning representation from textual and visual modalities jointly and acquire a common space where the features from different modalities are directly comparable [12, 1, 85]. In the task such as weakly supervised grounding also tries to find a common space for text and visual where we can find the correct region given a text query [88, 213, 214]. These works usually learn in a weakly supervised manner where human-annotated

image/video caption pairs were given. In our multimodal event coreference resolution task, we try to learn in a self-supervised manner where only the video and its ASR were given.

Text Event Extraction. Recognizing and extracting events in text is an important information extraction problem that has been thoroughly studied. Both document-level [215, 216] and sentence-level [217] methods have been proposed. Classic work by [218] and [219] leverage manually designed features for the task and formulate event extraction as a classification problem. More recent event extraction methods have leveraged neural models such as recurrent networks [197, 198], convolutional networks [220], graph networks [199, 221], joint neural model [222], conditioned generation [216] and transformers [200] to automatically learn task-relevant features.

A related line of work has focused on the problem of event argument extraction, where the goal is to predict event argument roles of entities in text to fill the roles of predicted event frames. [201] propose a hierarchical event argument extraction model leveraging modular networks to exploit argument role concept correlation. [202] propose a sampling-based method for jointly extracting events and arguments. Other methods have attempted to leverage zero-shot learning [223] and weak supervision [224] to further improve performance on both event and event argument extraction.

While impressive progress has been made in recent years, all of these methods exclusively focus on text and forego the oftentimes complex and complementary information found in visual media. In contrast, we propose to extract both events and event arguments from both text and video.

Visual event extraction. Event recognition has also been studied by the computer vision community, where it is commonly termed "situation recognition" [203, 206]. Analogous to textual event extraction methods, the goal of visual situation recognition is detecting events occurring in an image, the objects and agents involved, and identifying their roles. Most work in this space [203, 204, 205, 206] relies on the FrameNet [225] ontology derived from text which defines frames for each verb, along with semantic roles of arguments.

Seminal work by [203] introduced the *SituNet* dataset of images labeled with visual verbs and argument roles. Follow-up approaches have leveraged structured prediction mechanisms [204, 226] and attention [227] to further improve performance on *SituNet*. [206] extend *SituNet* with bounding

box annotations of event arguments and introduce a model for localizing event arguments in images. None of these target the video domain as we do or perform multimodal event extraction. More related to our work is [228], which introduces the video semantic role labeling dataset and task, where the target is to extract events and generate language description for arguments. Unlike [228], we propose to extract multimodal events and localize arguments, where components in the extracted event frame may appear in either modality.

Multimodal Event Extraction. Some prior work has leveraged multimodal information for text-only event extraction. [207] propose a method which learns to transfer visual knowledge from multimodal resources to text-only documents to improve event extraction. [208] supplement existing event detection benchmarks with image data and show significant performance gains by leveraging multimodal information for trigger disambiguation.

Most relevant to our work is [209]'s method which introduces the task of multimedia event extraction, where event frames are comprised of both visual and textual arguments. [209] leverage single-modality training corpora and weak supervision to train a cross-modal method, without any annotations. Our work has several important differences from [209]. First, we target the video modality, while [209] target images. This problem is significantly more challenging because video event extraction requires understanding the rich dynamics in videos. Additionally, because no datasets of video event argument role localization exist, we can not directly borrow existing image event extraction resources like [209]. Finally, we propose a novel multimodal transformer architecture for this task.

5.3 VM^{2}E^{2}Dataset

5.3.1 Dataset Collection

We introduce the VM^{2}E^{2}dataset which labels (1) Multimodal event coreference (2) Events and argument roles from 860 video article pairs.

Event types. The Linguistic Data Consortium (LDC) has created document-level event ontology based on previous LDC-supported ontologies ERE and ACE. These have been made publicly

Event Type	
CastVote (26)	Disaster.FireExplosion (60)
Contact.Broadcast (359)	Life.Injure (78)
Contact.Correspondence (75)	Justice.ArrestJail (31)
Contact.Meet (196)	ManufactureAssemble (44)
Conflict.Attack (147)	Movement.Evacuation(23)
Conflict.Demonstrate (242)	Movement.PreventPassage (43)
DamageDestroy (50)	Movement.Transport (287)
DetonateExplode (62)	Transcation.ExchangeBuySell (36)

Table 5.1: Event types in VM^2E^2. Numbers in parentheses represent the counts of visual events.

availble online[1]. The event types covered by the LDC ontology focus on issues related to disasters, attacks and activities from international news. We found that this ontology provides good coverage of many events found in world news and thus adopt it for our system. Because not all event types in the ontology are visually detectable, we manually selected event types defined in the LDC ontology that are: (1) Visually detectable: events that can be visually seen, and (2) Frequent: events that have a frequency > 20 in our dataset. This resulted in a set of 16 event types, which we show in Table 5.1. The full event type and argument role definition are included in the supplementary.

Candidate Video/Article Filtering. Given the 16 event types, we build a data collection pipeline. First, we use the event types and news source names as keywords to search on Youtube. We harvest from VOA, BBC, and Reuters. We choose these sources because we they are trustworthy and usually contain articles under the video such that the content is about the same event as the video. Second, we filter out videos that are longer than 16 minutes to avoid extra-long videos. Third, we check each video to make sure it contains at least one visual event. Starting from 1.2K videos, we end up with 860 video article pairs containing multimodal events. For the dataset, we will release the YouTube URLs that contains the video and article along with the annotations. We do not own the copyright of the video and the researcher shall use the data only for non-commercial research and educational purposes. More information about the Fair Use Notice will be included in the supplement.

[1] https://tac.nist.gov/tracks/SM-KBP/2018/ontologies/SeedlingOntology

5.3.2 Dataset Annotation Procedure

In order to collect annotations, we perform the following steps for videos and text. First, annotators watch the entire video to identify all event instances in the video. Next, for each event instance, the temporal boundary, event type, and co-referential text event (if existent) is annotated and three keyframes within the temporal boundary are selected. Then, for each selected keyframe all arguments are identified. Finally, for each argument, the argument role type, entity type, and co-referential text event (if existent) is annotated. We extensively annotate the videos and sampled keyframes with bounding boxes for argument roles to ensure none are missed.

5.3.3 Annotation interface

Our data annotation interface for video is shown in Figure 5.2. Each annotator needs to walk through the whole video and corresponding articles. As shown in the figure, we have a list of event types for the annotators to label the start time and end time. The same event can appear multiple times in the same video. We also allow overlap between different events.

Event Type	**Argument Role**
CastVote	Voter,Candidate,Ballot,Result,Place
Contact.Broadcast	Communicator, Recipient, Instrument, Topic, Place
Contact.Correspondence	Participant, Instrument, Topic, Place
Contact.Meet	Participant, Topic, Place
Conflict.Attack	Attacker, Target, Instrument, Place
Conflict.Demonstrate	Demonstrator, Demonstrator, VisualDisplay, Topic, Target, Place
DamageDestroy	Damager, Artifact, Instrument, Place
DetonateExplode	Attacker, Target, Instrument, ExplosiveDevice, Place
Disaster.FireExplosion	FireExplosionObject, Instrument, Place
Life.Injure	Victim, Injurer, Instrument, BodyPart, MedicalCondition, Place
Justice.ArrestJail	Jailer, Detainee, Crime, Place
ManufactureAssemble	ManufacturerAssembler, Artifact, Components, Instrument, Place
Movement.Evacuation	Transporter, PassengerArtifact, Vehicle, Origin, Destination
Movement.PreventPassage	Transporter, PassengerArtifact, Vehicle, Preventer, Origin, Destination
Movement.Transport	Transporter, PassengerArtifact, Vehicle, Origin, Destination
Transcation.ExchangeBuySell	Giver, Recipient, AcquiredEntity, PaymentBarter, Beneficiary, Place

Table 5.2: Event types and argument roles in VM^{2}E^2.

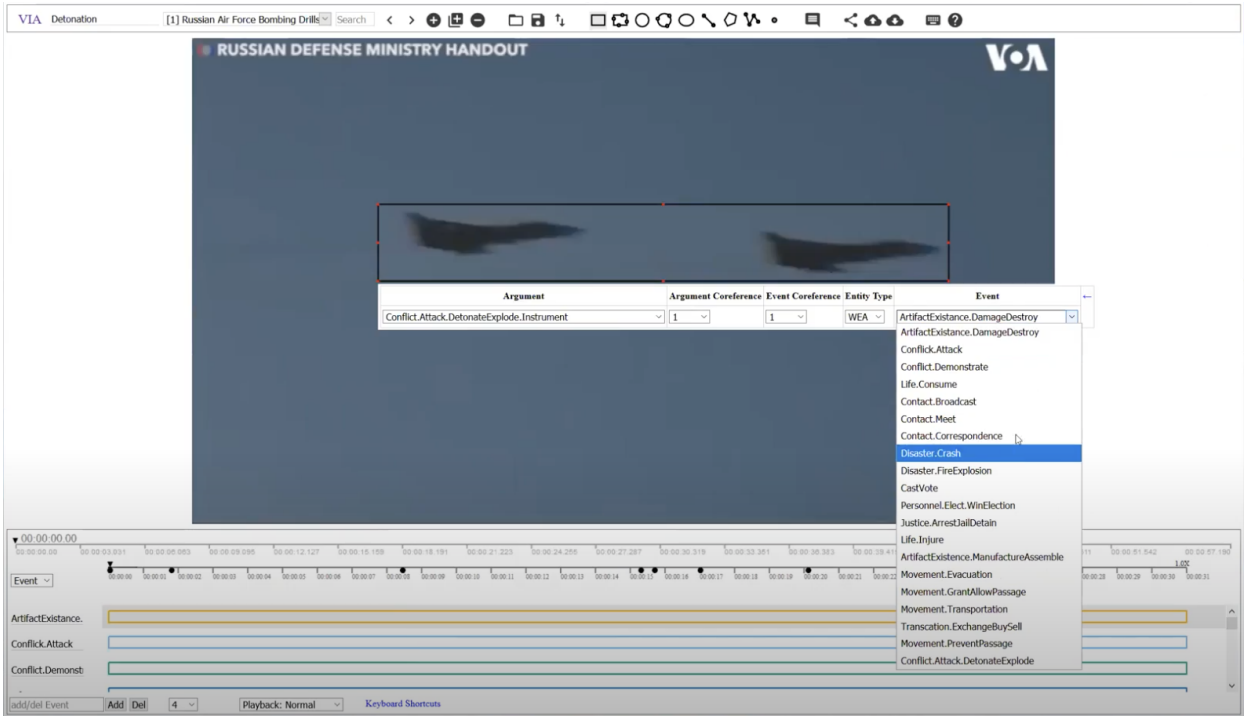


Figure 5.2: Annotation interface of the video. We annotate the event temporal of each video event. Also, we will annotate the multimodal event coreference between the video event and text event. For the argument role, we select 3 frames to annotate the bounding box.

5.3.4 Event type

The event type along with its argument roles are shown in Table 5.2. We followed The Linguistic Data Consortium (LDC) ontology defined for the AIDA program. These have been made publicly availble online[2].

5.3.5 Event Proposal Generation

To acquire the temporal boundary of the video event, we use the Boundary Sensitive Network [229] for temporal proposal generation in the video clips. We fine-tune the network with the VM^{2}E^{2}training set to better capture the action semantics within the dataset. Table 5.3 shows the proposal generation results for VM^{2}E^{2}test set. Similar to [229, 230], we evaluate the improvement in the ability of BSN to generate proposals which have high temporal overlap with ground truth

[2] https://tac.nist.gov/tracks/SM-KBP/2018/ontologies/SeedlingOntology

proposals. To quantify this improvement, we measure the recall (AR) over multiple temporal-IoU thresholds (0.5 to 0.95 with an increment of 0.05) for a fixed number of proposals(N). We also measure the area under(AUC) average recall(AR) at different number of proposals(N) curve.

Although we use ground truth proposals for event extraction and argument role labelling section of the experiments in the current work, our method can be extended to work with automatically generated proposals. Hence, our method combined with any proposal generation technique, can be considered as an end-to-end solution to multimedia event extraction given a video-article pair.

Training	**AR**		**AUC**
	@1	@100	
ActivityNet	0.11	0.52	38.52
ActivityNet + VM^2E^2	0.18	0.67	54.94

Table 5.3: Fine-tuning the BSN pipeline with VM^2E^2shows significant improvement in proposal generation and retrival performance.

5.3.6 Quality control

We train fourteen NLP and computer vision researchers to complete the annotation work with two independent passes. After annotation, two expert annotators perform adjudication. For the multimodal event coreference resolution, we sampled 10% of annotations and reached an Inter-Annotator Agreement (IAA) of 84.6%. For the event and argument role labeling, we sampled 10% of annotations and reached an Inter-Annotator Agreement (IAA) of 81.2%.

Type	**1-to-1**	**1-to-n**	**n-to-1**	**n-to-n**	**Total**
Count	202	104	260	286	852

Table 5.4: Multimodal event coreference link types found in VM^2E^2.

Document		**Event Mention**		**Argument Role**	
Sentence	Video	Textual	Visual	Textual	Visual
13,239	860	4,164	2,702	18,880	5,467

Table 5.5: Annotated VM^2E^2 data event and argument role statistics.

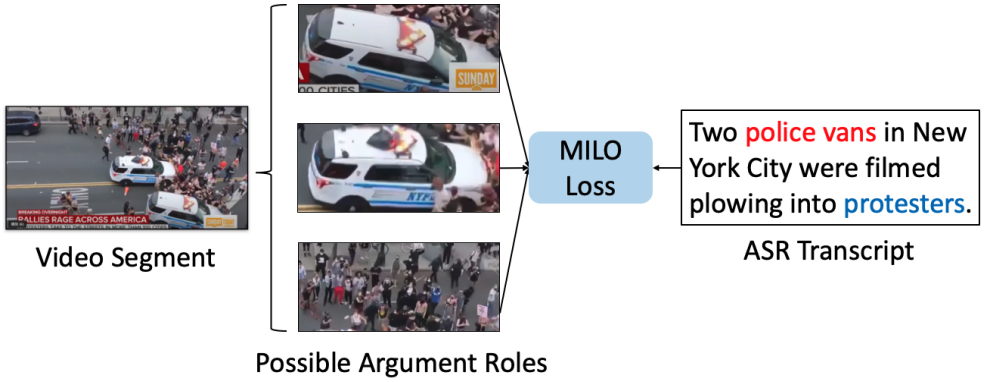


Figure 5.3: Self-supervised multimodal event coreference resolution by considering the possible argument roles that participate in the event.

5.3.7 Dataset statistics

Overall, we annotated 852 multimodal event coreference links between video segments and sentences. Table 5.4 breaks down the annotations into relation categories: **1-to-1**, where one text event is only coreferential with a single video event, and **n-to-n**, where multiple text events and video events are corerefential. We also provide data statistics for the event extraction and argument role annotations in Table 5.5.

5.4 Method

5.4.1 Problem Formulation

In Multimodal Event Coreference Resolution, given M sentences and N video segments in a multimedia document, the system is required to predict the coreference $c_{ij} \in \{0, 1\}$ between a sentence x_i and a video segment y_j. In Joint Multimodal Event Extraction and Argument Role Labeling, given a text sentence x_i and a video segment y_j, the system is required to predict the multimodal event type e, the text mention t_e, the text mention t_{a_k} and the bounding box $bbox_{a_k}$ for each argument role a_k.

5.4.2 Multimodal Event Coreference Resolution

We aim to learn a common space across the video and text modalities such that the embeddings across these modalities are close if they represent the same event. This is a particularly challenging task since in an unannotated multimodal document, we don't know which video segment aligns with which article sentence. Inspired by multimodal self supervised methods learning from instructional videos [12], we learn the common space across the two modalities from our unannotated video clips using their auto-generated ASR transcripts as supervision. To accomplish this, we use a standard noise contrastive loss (NCE) [231] $\mathcal{L}_{NCE}$:

$$\max_{f,g} \sum_{i=1}^{n} \log \left(\frac{e^{f(x_i)^\top g(y_i)}}{e^{f(x_i)^\top g(y_i)} + \sum\limits_{(x',y')\sim\mathcal{N}_i} e^{f(x')^\top g(y')}} \right)$$

where x represents a sentence and y a video clip. f and g are the two learnable networks that project the two features into a common space. The loss learns to pull the positive pairs (x_i, y_i) that co-occur in time while pushing mis-matched pairs in the batch away.

Additionally, we find the region information (arguments that participate in the event) to be crucial in finding coreferential events between video and text. For example, when we see an *Attack* event in the text, we might find the objects "`van`" or "`protester`" in the video to be important since they participate in the event as shown in Figure 5.3. In order to learn such correspondences between text and object regions, we introduce the Multi-Instance Learning from Objects $\mathcal{L}_{MILO}$ loss:

$$\max_{f,h} \sum_{i=1}^{n} \log \left(\frac{\sum\limits_{(x,z)\in\mathcal{P}_i} e^{f(x)^\top h(z)}}{\sum\limits_{(x,z)\in\mathcal{P}_i} e^{f(x)^\top h(z)} + \sum\limits_{(x',z')\sim\mathcal{N}_i} e^{f(x')^\top h(z')}} \right)$$

where z represents the regions in the video clip and h is a projection layer. Given a specific video instance i, $\mathcal{P}_i$ represents the *positive* region/sentence candidate pairs (i.e. the region and sentence co-occur in time, see Figure 5.3) while $\mathcal{N}_i$ represents the set of negative region/narration pairs that were sampled from different time frames. The learning objective takes all possible region

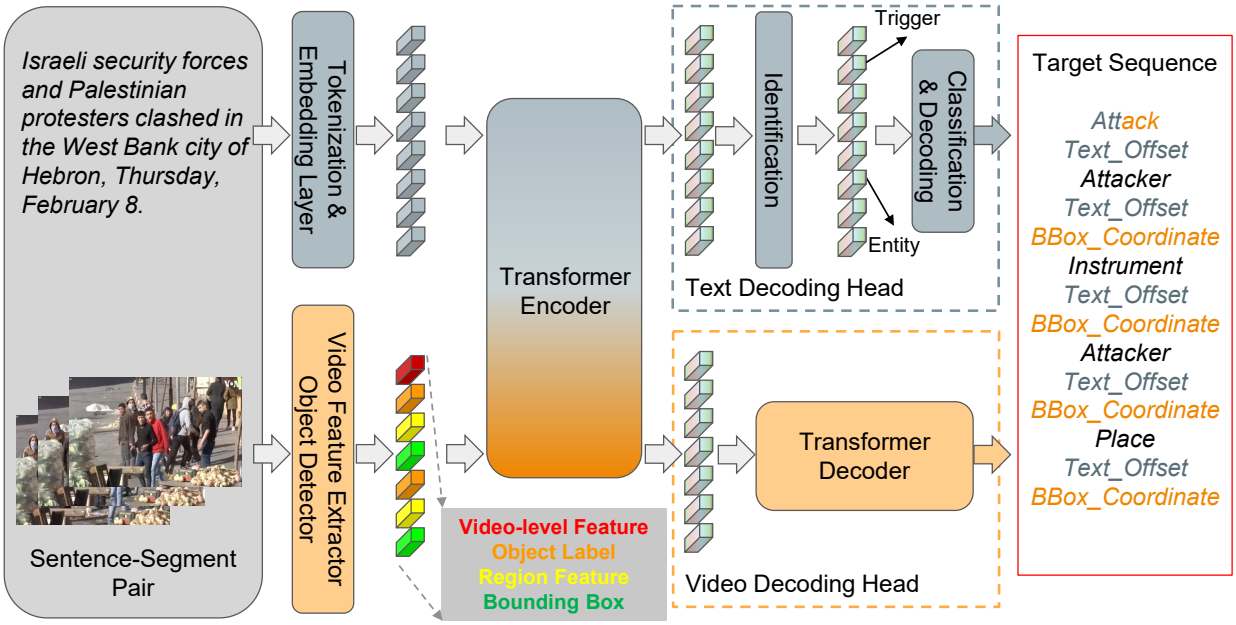


Figure 5.4: Multimodal transformer for joint event extraction and argument role labeling. In the target sequence, blue-gray and light orange are for textual and visual decoding heads, respectively.

information into consideration by summing over all the pairs. The model learns in a multi-instance fashion to select the regions that are most important for multimodal event coreference resolution. Our final multimodal coreference loss combines both global and local constraints:

$$\mathcal{L}_{mmcoref} = \mathcal{L}_{NCE} + \mathcal{L}_{MILO} \quad .$$

5.4.3 Joint Multimodal Event Extraction and Argument Role Labeling

Inspired by recent work [232] on leveraging multimodal transformers to jointly process text and visual information, we propose a joint multimodal transformer (JMMT) to extract events and arguments from a paired text sentence and video clip. The proposed JMMT has an encoder-decoder structure: the encoder extracts and fuses information from both modalities (text and video), while the decoder is more complex. The decoder consists of two heads: one for detecting trigger words, event types, and arguments from text, and the other for classifying video event types and predicting bounding boxes for visual arguments. With this joint encoder, JMMT can effectively leverage

contextual information to extract events and label argument roles.

As shown in Figure 5.4, JMMT takes text and visual tokens as input. For text tokens, we follow [233] to embed text tokens. For visual tokens, we use four feature types to comprehensively represent both global and local information: 1) video-level features extracted from the whole video segment capture the global event context; 2) frame-level object labels produced by an object detector; 3) frame-level region features extracted from bounding boxes detected by the object detector provide fine-grained argument information; 4) frame-level object coordinates also provided by the object detector for localization of arguments. Note that we sample t frames and for each frame, we sample k objects with the highest confidence scores. The text and visual tokens are then stacked as a sequence and input to the encoder for joint processing.

Our encoder and decoder are initialized from transformers pretrained on text corpora [233]. Our decoding head for text event extraction is borrowed from [234]'s state-of-the-art text event extraction model. For text decoding, we take encoder outputs as input and first merge these multimodal contextualized embeddings of word pieces to obtain a representation for each word in the input sequence. Then we process these word representations for identification, classification and decoding, following [234]. For the video decoding head, we leverage the decoder from [233]'s pretrained text transformer and cast the task as a sequence-to-sequence prediction problem. We set the target sequence as $\{e, a_1, bbox, a_2, bbox, ..., a_n, bbox\}$, which begins with event type e and then goes through each argument role a_i to produce the bounding box coordinates $bbox$ on the sampled key frames.

Each decoding head is supervised by its own loss term and the gradients are both back-propagated to the encoder. The text decoding head is supervised based on the objective $\mathcal{L}_{text}$ proposed in [234] and the video decoding head is trained using a standard teacher-forcing strategy with cross-entropy loss [233] $\mathcal{L}_{video}$. The overall objective is

$$\mathcal{L}_{JMMT} = \mathcal{L}_{text} + \mathcal{L}_{video}.$$

Input	Model	Text Evaluation						Video Evaluation						Multimedia Evaluation					
		Event			Mention			Argument			Role			Event			Mention		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Text	OneIE	38.5	52.1	44.3	16.6	21.8	18.8	-	-	-	-	-	-	38.5	52.1	44.3	16.6	21.8	18.8
Video	JSL	-	-	-	-	-	-	24.1	17.1	20.0	2.2	2.8	2.4	24.1	17.1	20.0	2.2	2.8	2.4
	$\text{JMMT}_{\text{Video}}$	-	-	-	-	-	-	26.6	29.2	27.8	8.9	10.1	9.5	26.6	29.2	27.8	8.9	10.1	9.5
Multimedia	WASE	33.6	53.8	41.4	15.2	22.1	18.0	20.4	14.0	16.6	2.8	1.3	1.7	34.0	54.0	41.8	15.3	22.1	18.1
	JMMT	39.7	56.3	**46.6**	17.9	24.3	**20.6**	32.4	37.5	**34.8**	9.2	10.6	**9.9**	41.2	56.3	**47.6**	18.8	24.7	**21.3**

Table 5.6: Event and argument extraction results (%). We evaluate three categories of models in three evaluation settings. By jointly leveraging multimodal context, JMMT significantly improves multimedia event extraction from video segments and sentences.

In this way, the proposed JMMT can effectively fuse multimodal information and jointly extract events and arguments.

5.5 Experiments

5.5.1 Dataset

Event coreference resolution. Our model is trained on our unannotated dataset, which contains 3K videos and corresponding automatically generated speech transcriptions. We test our model on the annotated dataset, which contains 860 videos and their articles from YouTube.

Event extraction and argument role labeling. We split the annotated 860 video-article pairs into 645 and 215 for training and testing, respectively. To focus on joint multimedia event extraction, we sample all the coreference segment-sentence pairs for training and evaluation.

5.5.2 Evaluation Setting

Event coreference resolution. We evaluate our model on the annotated event coreference data by predicting whether every possible sentence-video segment pair from the same multimodal document is coreferential or not. We perform feature similarity between the text and video features within the learned joint space and predict the pair as coreferential if their similarity surpasses a threshold. We adopt traditional link prediction metrics, i.e. precision, recall, F_1, and accuracy for evaluation.

Event extraction and argument role labeling. We evaluate models on text-only, video-only, and

multimedia event mentions in the VM^{2}E^{2}dataset. We follow the common event extraction metrics, i.e. precision, recall, and F_1. For a text event mention, we follow [209] to only consider it as correct if its trigger offsets and event type both match a reference trigger. Similarly, a textual argument is only considered as correct when its offsets, event type, and role type all match a reference argument. Analogously, a video event mention is considered correct if its segment and event type match a reference segment. A video argument is considered correct if its localization, event type and role type matches a reference argument. A visual argument is correctly localized if its Intersection over Union (IoU) with the ground truth bounding box is greater than 0.3. Finally, a multimedia event mention is considered correct if its event type and trigger offsets (or the video segment) match a reference trigger (or the reference segment). Arguments of multimedia events with either a correct textual or visual argument mention are considered correct.

5.5.3 Baseline methods

Event coreference resolution. We compare our method against several self-supervised models that learn a joint visual text space. Specifically, HowTo100m [12] learn a joint video-text space using a max-margin ranking loss [235]. NCE loss [231] trains a classifier to discriminate between real instances and a generated noise distribution. MIL-NCE [1] further extends NCE by explicitly considering the misalignment of the video segment and ASR transcript to design a multi-instance loss. We do not compare to retrieval methods that require fine-tuning.

Event extraction and argument role labelling. 1) **Text-only baseline:** We re-implement a state-of-the-art method, OneIE [234]. For a fair comparison, we use the same text encoder [233] as our JMMT. 2) **Video-only baseline:** As no existing method addresses the problem of event extraction and argument role labeling from videos, we adopt the state-of-the-art method for grounded image event extraction, JSL [206], to extract events and arguments from each annotated key frame. 3) **Multimedia baseline:** As previous multimedia event extraction methods only consider image-text pairs, we borrow one of the best performing models on M^{2}E^2 [209], WASE [209], as our baseline for multimedia event extraction. Note that we rebuild WASE to extend from its ontology to our

VM^{2}E^{2}Event Type	SWiG Verb Class
CastVote	Voting
Contact.Broadcast	Speaking
Contact.Correspondence	Calling, Dialing, Phoning, Telephoning
Contact.Meet	Communicating, Interviewing, Talking, Discussing, Shaking
Conflict.Attack	Attacking, Punching, Kicking, Striking, Shooting
Conflict.Demonstrate	Protesting, Marching, Displaying, Gathering
DamageDestroy	Breaking, Destroying
DetonateExplode	-
Disaster.FireExplosion	Flaming, Erupting, Burning
Life.Injure	-
Justice.ArrestJail	Detaining, Restraining, Arresting
ManufactureAssemble	Assembling
Movement.Evacuation	-
Movement.PreventPassage	Blocking, Guarding
Movement.Transport	Driving, Boating, Disembarking, Landing, Piloting, Steering, Taxiing, Commuting, Riding, Boarding, Biking
Transaction.ExchangeBuySell	Paying, Selling

Table 5.7: Mapping used to convert the SWiG verbs to VM^{2}E^{2}events. Note that 3 events do not have any mapping. We do not evaluate the JSL baseline over these events.

ontology. SWiG (Situations with Grounding) dataset provides the annotations corresponding to the visually groundable verbs and the nouns associated with them. To evaluate the JSL[234] model on VM^{2}E^{2}dataset, we map the SWiG verb classes onto the VM^{2}E^{2}event classes as described in Table 5.7. Note that some classes in SWiG do not have any verb corresponding to the VM^{2}E^{2}event. Hence, these events are never predicted by the JSL model. For fair comparison, we calculate the precision and recall with respect to the remaining classes only. In a similar manner, we reformulate the mappings used in WASE [209] to extend the ontology of Image M^{2}E^2[209] to the VM^{2}E^{2}ontology and retrain WASE as our baseline.

5.5.4 Implementation details

For the visual branch of the multimodal event corerefence resolution model we follow [12] and use pre-trained 2D features from a ResNet-152 model [71] trained on ImageNet [72] and 3D features from a ResNeXt-101 model [73] trained on Kinetics [51]. For the textual branch, a GoogleNews

pre-trained Word2vec model [74] provides word embeddings, followed by a max-pooling over words in a given sentence to extract a sentence embedding. We use Faster R-CNN [236] pre-trained on the Visual Genome dataset [237] as our object detector. For selecting the number of objects, we sort by the confidence score of each object and select the top 5 as possible argument roles. Also, we uniformly sample 3 frames in each video segment and end up with 15 objects for each segment. For each feature extraction branch (text, video, object), we apply separate fully-connected layer and a gated unit for projection to common space. We use an Adam optimizer [75] with a learning rate of 1e−4. The batch size is set to 256 video clips. The model is trained for 50 epochs on one NVIDIA TITAN RTX for about 2 hours. We further split the 860 video data into 200 video article pairs for the validation set and 660 for testing the performance. The parameter search for the threshold was done in the validation set by selecting the highest F1 score, and the similarity score above 0.13 will be viewed as positive pairs for prediction.

For event extraction and argument role labeling, we use the same video-level feature and object detector. We use T5-base [233] with pre-trained weights provided in HuggingFace [238] for initialization. For video-level features and region features, we separately use a fully-connected layer to project them into 768-D space to be aligned with text embeddings. We directly use text embedding layer to embed bounding box coordinates. For the text decoding head, we borrow implementations from the official implementation[3] of OneIE and use the same hyper-parameters. The video decoding head uses Beam Search for decoding in inference, with a beam width of 5. During training and evaluation, we sample annotated $t = 3$ frames and extract $k = 15$ objects for each frame. We use a batch size of 6 examples per GPU, and distribute the training over 4 NVIDIA V100 GPUs. We use Adam with a learning rate of 1e−4 to optimize our models. We train our models for 150 epochs.

[3] http://blender.cs.illinois.edu/software/oneie

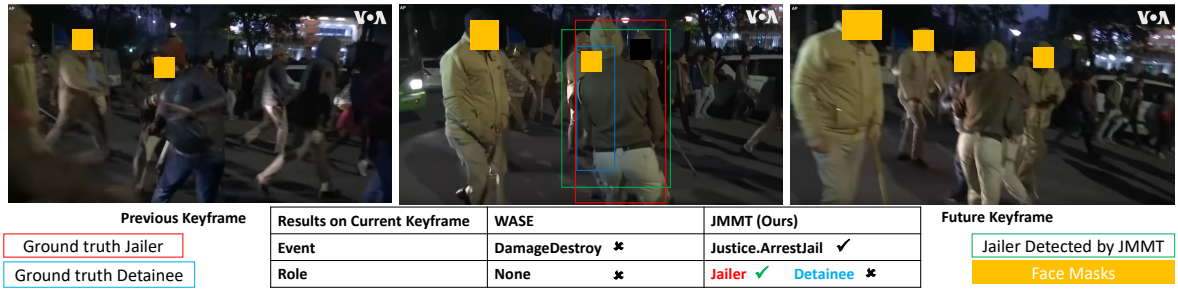


Figure 5.5: Visualization of event extraction results on one video segment. We mask faces (orange boxes) for privacy.

Method	Visual Model	TR	P	R	F_1	Acc
HowTo100M	R152+RX101	N	32.2	62.8	44.3	55.2
NCE	R152+RX101	N	35.5	68.3	45.5	47.5
Ours	R152+RX101	N	**38.4**	**76.4**	**51.5**	**59.6**
MIL-NCE	S3D-G	Y	37.8	75.0	50.6	59.2

Table 5.8: Multimodal event coreference resolution results. Our method outperforms all baselines, including one with a more powerful and trainable visual backbone (indicated by TR).

5.5.5 Quantitative Performance

Event coreference resolution. We first examine the results of the multimodal event coreference resolution task in Table 5.8. All the methods we compare share the same text feature extractor. For visual feature extraction, HowTo100M, NCE, and our method apply ResNet-152 (R152) and ResNeXt-152 (RX101) followed by [12]. MIL-NCE uses a more advanced video feature extraction backbone, S3D-G [19]. Our model significantly outperforms all previous methods using the same architecture, as well as those models with a trainable (TR) and more powerful visual backbone [1].

Event extraction and argument role labeling. The proposed JMMT significantly improves the event extraction performance over baseline methods as shown in Table 5.6. Compared to text-only OneIE or video-only JSL baselines, the JMMT produces at most **74%** relative gain in event extraction, which demonstrates the importance of leveraging multimodal information for understanding complex events. Compared to previous methods on image-text multimedia event extraction, the superior performance of JMTT verifies 1) the effectiveness of the powerful transformer model for multimodal information fusion and 2) the importance of modeling dynamics

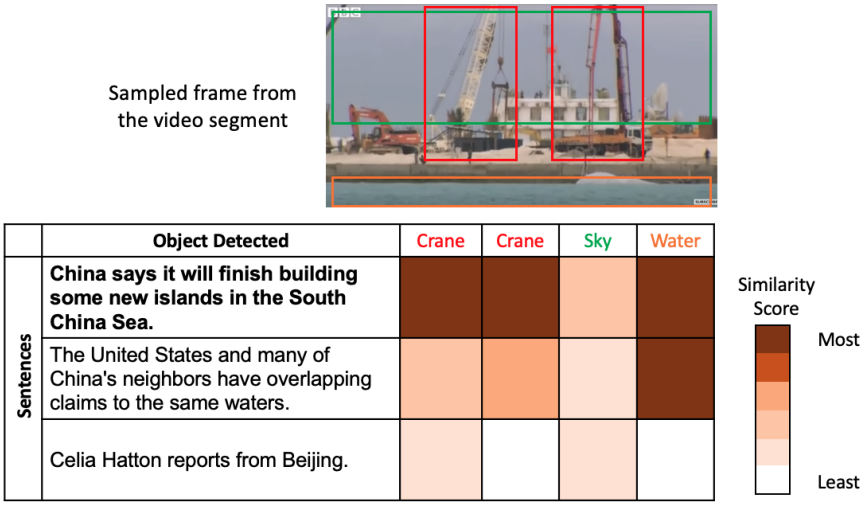


Figure 5.6: Event coreference resolution visualization. The bold sentence is correctly selected as coreferential within the article by the model.

in videos.

Method	Supervision	Video Evaluation					
		Event Mention			Argument Role		
		Precision	Recall	F1	Precision	Recall	F1
JSL	Supervised	24.1	17.1	20.0	2.2	2.8	2.4
JMMT	Supervised	26.6	29.2	27.8	8.9	10.1	9.5
MCN	Self-supervised	28.4	19.2	19.5	2.1	2.4	2.2

Table 5.9: Self-supervised event extraction and argument role labeling.

Self-supervised event extraction and argument role labeling. Instead of utilizing supervised annotation as the JMMT model shown above, we explore the multimodal common space by video-to-ASR text self-supervision to train our Multimodal Clustering Network proposed in chapter 2. As shown in Table 5.9, result demonstrates that we achieved comparable results with the SOTA transformer model in event extraction tasks. We assume the performance drop in argument role labeling is due to the lack of bounding box supervision, which is not provided in the self-supervision model.

5.5.6 Qualitative Analysis

We visualize results from our event coreference resolution model in Fig 5.6. We observe that the model correctly selects the most appropriate sentence for a given video segment. Also, we find that the model learns to associate object regions to the words in the sentence. For example, the first sentence had a high similarity score with the object *'Crane'* since it mentioned *'building some new island'*.

We also visualize results of event extraction. As shown in Fig. 5.5, the center frame is very hard to extract events from due to the occlusion of arguments. With only this image as input, WASE fails to extract events and arguments. However, our JMMT successfully recognizes the event and detects "Jailer" in the image with the help of video-level dynamics and the context of the previous and next frames. This example illustrates both the importance and difficulty of multimedia event extraction from videos and articles. We also observe that our JMMT fails to recognize the "Detainee" because of occlusion. This indicates the possibility of leverage entity tracking to further improve VM^2E^2 which we leave as future work.

5.6 Limitation

Dataset. Our method was based on our collected dataset, which might contain unintended societal, gender, racial, and other biases when deploying models trained on this data. Also, our problem formulation assumes the video and article are about the same topic. This assumption leads our method to work on news videos and instructional videos. If we didn't constrain the videos to these genres, we might collect videos without articles or videos with unrelated articles such as music videos and animated videos on YouTube.

Evaluation. Our proposed pipeline could be combined as a two-step approach, starting from raw videos and articles and then acquiring both modalities' event and argument roles. However, in our evaluation, we only evaluate the argument roles on annotated keyframes since we don't have the annotation for every video frame due to the expense of annotation. End-to-end evaluation

for all frames is not practical because the chosen frames from the multimodal event coreference resolution model are not guaranteed to be the ground truth frames on which we have annotations. Consequently, we cannot evaluate the predictions of our multimodal event extraction model (stage 2) when we use predicted frames as input since we do not have annotations on the frames on which predictions are made (thus, the results in those frames could be correct or incorrect).

5.7 Summary

In this chapter, we have introduced a novel task VM^2E^2- given a video with a paired article, our first goal is to find coreferenced events across modalities. Also, our task requires extracting the event type and argument roles from both modalities. 860 video-article pairs were labeled to support this task. We developed a novel self-supervised multimodal network that learns a common embedding space by processing local (object region) and global (video level) semantic relationships to perform multimodal event coreference resolution. In addition, we present a new architecture JMMT that jointly extracts events and arguments from both modalities using an encoder-decoder-based multimodal transformer. Our extensive experiments on multiple settings show that considering region information and a joint transformer for both modalities is essential for good performance on the two subtasks in VM^2E^2. Our dataset collection pipeline and approach can be extended to more scenarios such as instructional videos and other videos that contain video-article pairs for extracting multimodal events across both modalities.

Chapter 6: Conclusion

6.1 Summary of Contributions

In this thesis, we present various contributions to improve representation learning performance from unlabelled multimodal data. Our main focus was on Learning Representations with Multi-Modal Self-Supervision. Our key contribution, MCN (Chapter 2), served as the foundation for the majority of the work in this thesis, where we explored the importance of using three modalities, audio, video, and text, in large-scale data pretraining for multimodal common space. Additionally, we used a combination of noise-contrastive (NCE) training and clustering objectives for multimodal self-supervision. The success of this approach in MCN inspired us to explore each of these axes further in subsequent chapters, such as spatio-temporal grounding (Chapter 3) where we saw similar gains in performance on the video-text retrieval task. Overall, our work has inspired a few other works in the literature to leverage large-scale multimodal video data to improve common space representations for further zero-shot settings [24, 27], where human annotation is not needed throughout the pipeline.

Besides learning to represent video by a single vector, we have presented a novel approach for multimodal representation learning by combining single vector representation learning with token-level (video region and words) representation learning, which we have called spatio-temporal grounding. Our work on MCN (Chapter 3) highlighted the need for finer granularity information from both visual and textual features in multimodal representation learning. While previous works have focused on learning a single vector representation for a video segment or sentence, our approach accounts for the spatial and temporal dimensions of the video, making it better suited for modeling complex human-object interactions. Through our experimentation and evaluation, we have shown that this approach improves representation learning performance for instructional

videos. We believe that this spatio-temporal grounding approach will be an important component in future multimodal representation learning research, particularly in the context of instructional videos.

We have explored ways to improve the performance of contrastive learning for representation learning from video data. Specifically, we have examined how incorporating tracking supervision and an attention loss can help alleviate the strong instance discrimination assumption in the sampling strategy. Our findings, presented in chapter 5, demonstrate the effectiveness of this approach in improving video representation learning and reducing the bias towards background information. We have also shown that this approach requires unsupervised video object tracking to be implemented. Overall, our research in this area has the potential to make a significant impact on the field of representation learning from video data.

In the end, we proposed a new task Video Multimodal Event Extraction (VideoM2E2) which motivated the need for multimodal common space from multimodal self-supervised learning. By training video-text representations with video and article pairs on Youtube videos, we showed that this approach can improve performance compared to single-modality training. Furthermore, we demonstrated that visual and textual information can serve as reciprocal information without specifically paired supervision. To encourage further research in this area, we have publicly released a new dataset, Video M2E2, which provides event and argument role annotation from both video and article pairs in Youtube News videos. Multimodal event extraction is an exciting research direction as it can open up opportunities to train better video-text representations and make video and text content on the Internet more accessible and comprehensive to a wider population.

6.2 Open Issues

Here we present a few areas that are exciting directions for our work:

Training with more modalities. In this thesis, we have investigated the use of pairs of modalities, specifically video-audio and video-text, for multi-modal self-supervised representation learning. Previous research has utilized combinations of video, audio, and text modalities for this purpose [3,

2]. However, we believe there is potential to further improve representation learning by incorporating additional modalities such as optical flow and depth. These added modalities have the potential to capture different semantic features of the input, leading to a better representation if the model can effectively correlate the signals. However, it is also challenging to learn from an excessive number of modalities due to differences in semantic strengths, granularity, and learning speeds. Furthermore, there are technical challenges in training multiple modality encoders end-to-end using GPU memory. Advances in chip development may provide a solution to this problem.

Training with absent modalities. In this thesis, we have investigated the use of three modalities, video, audio, and text, for multi-modal self-supervised representation learning. However, in certain scenarios, some modalities may be absent. For instance, some videos may not have accompanying audio or the audio may be irrelevant noise and background music. Additionally, in certain languages, the automatic speech recognition (ASR) system may not be well-developed, resulting in missing the text modality or inaccurate text transcripts [96]. Investigating the training of multi-modal representations in the presence of missing modalities is an exciting new direction that aims to construct a more robust system that can adapt to different inputs. This research direction also aims to explore how different modalities contribute to the overall performance of the learned representations [27], or develop a more intelligent way of weighting useful modalities for learning better representations.

Unpaired training. In this thesis, we demonstrate the effectiveness of utilizing paired video segments and automatic speech recognition (ASR) text for self-supervised contrastive learning. However, a more challenging setting is to learn from unpaired data, where there is no direct correlation or distance supervision between the video and text [239]. To tackle this problem, previous works have attempted to retrieve video-text pairs as pseudo pairs, while others aim to leverage weaker supervision, such as using long video-article pairs instead of video segment-to-sentence pairs.

Joint Video and Image representations. We have examined the potential of using video data for representation learning in this thesis. The temporal dimension of video provides multiple viewpoints

of objects, and its rich multimodal information offers the opportunity to learn highly semantic representations. However, previous research has not yet succeeded in achieving state-of-the-art performance in the image domain using video-based representations. Learning joint image and video representations from video data that can be applied to both image and video tasks is still a significant challenge. However, recent developments in transformer architecture may provide a solution as it allows for consistent tokenization of image and video data, enabling mixed dataset training.

Learning from multimodal for multilingual understanding. In this thesis, we have focused on the video data with English Automatic Speech Recognition (ASR) text. However, there is a significant potential to expand our research to other languages, particularly those with low resources [240]. One potential avenue for exploration is to use visual data as a universal language while learning in a multi-lingual setting. This approach could potentially enhance performance on lower resource language datasets by pre-training on large-scale English datasets.

References

[1] A. Miech, J. Alayrac, L. Smaira, I. Laptev, J. Sivic, and A. Zisserman, "End-to-end learning of visual representations from uncurated instructional videos," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 9876–9886.

[2] J. Alayrac *et al.*, "Self-supervised multimodal versatile networks," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, 2020, pp. 25–37.

[3] H. Akbari *et al.*, "Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[4] Y. M. Asano, C. Rupprecht, and A. Vedaldi, "Self-labelling via simultaneous clustering and representation learning," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, OpenReview.net, 2020.

[5] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 9726–9735.

[6] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 1597–1607.

[7] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, PMLR, 2021, pp. 8748–8763.

[8] P. Chen *et al.*, "Rspnet: Relative speed perception for unsupervised video representation learning," in *AAAI Conference on Artificial Intelligence*, 2021.

[9] M. Patrick *et al.*, "On compositions of transformations in contrastive self-supervised learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[10] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 4724–4733.

[11] J. F. Gemmeke *et al.*, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2017, pp. 776–780.

[12] A. Miech, D. Zhukov, J. Alayrac, M. Tapaswi, I. Laptev, and J. Sivic, "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 2630–2640.

[13] M. Bain, A. Nagrani, G. Varol, and A. 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*, 2021, pp. 1728–1738.

[14] J. Y. Kim *et al.*, "A comparison of online automatic speech recognition systems and the nonverbal responses to unintelligible speech," *arXiv preprint arXiv:1904.12403*, 2019.

[15] F. Filippidou and L. Moussiades, "A benchmarking of ibm, google and wit automatic speech recognition systems," in *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part I 16*, Springer, 2020, pp. 73–82.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 1998.

[17] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 5297–5307.

[18] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[19] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XV*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., ser. Lecture Notes in Computer Science, vol. 11219, Springer, 2018, pp. 318–335.

[20] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 6450–6459.

[21] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *2015 IEEE International Conference on*

Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, IEEE Computer Society, 2015, pp. 4489–4497.

[22] É. Grave, "A convex relaxation for weakly supervised relation extraction," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar: Association for Computational Linguistics, 2014, pp. 1580–1590.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. 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)*, Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186.

[24] A. Rouditchenko *et al.*, "Avlnet: Learning audio-visual language representations from instructional videos," in *Interspeech*, 2021.

[25] D. Harwath, A. Recasens, D. Surís, G. Chuang, A. Torralba, and J. Glass, "Jointly discovering visual objects and spoken words from raw sensory input," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 649–665.

[26] A. Vaswani *et al.*, "Attention is all you need," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon *et al.*, Eds., vol. 30, 2017, pp. 5998–6008.

[27] N. Shvetsova *et al.*, "Everything at once-multi-modal fusion transformer for video retrieval," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 20 020–20 029.

[28] *Project webpage*, https://www.di.ens.fr/willow/research/howto100m/, 2019.

[29] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE, vol. 1, 2005, pp. 539–546.

[30] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *AISTATS*, 2010.

[31] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," 2018.

[32] P. Goyal, D. Mahajan, A. Gupta, and I. Misra, "Scaling and benchmarking self-supervised visual representation learning," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 6390–6399.

[33] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *ArXiv preprint*, 2012.

[34] H. Kuehne, H. Jhuang, E. Garrote, T. A. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011*, D. N. Metaxas, L. Quan, A. Sanfeliu, and L. V. Gool, Eds., IEEE Computer Society, 2011, pp. 2556–2563.

[35] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9640–9649.

[36] J. Xu, T. Mei, T. Yao, and Y. Rui, "MSR-VTT: A large video description dataset for bridging video and language," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 5288–5296.

[37] X. Wang, J. Wu, J. Chen, L. Li, Y.-F. Wang, and W. Y. 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*, 2019, pp. 4581–4591.

[38] L. Zhou, C. Xu, and J. J. Corso, "Towards automatic learning of procedures from web instructional videos," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[39] H. Alwassel, D. Mahajan, B. Korbar, L. Torresani, B. Ghanem, and D. Tran, "Self-supervised learning by cross-modal audio-video clustering," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[40] A. J. Piergiovanni, A. Angelova, and M. S. Ryoo, "Evolving losses for unsupervised video representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 130–139.

[41] Y. Aytar, C. Vondrick, and A. Torralba, "See, hear, and read: Deep aligned representations," vol. abs/1706.00932, 2017.

[42] L. Kaiser *et al.*, "One model to learn them all," vol. abs/1706.05137, 2017.

[43] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *AISTATS*, 2010.

[44] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. of CVPR*, IEEE, 2006.

[45] Y. M. Asano, M. Patrick, C. Rupprecht, and A. Vedaldi, "Labelling unlabelled videos from scratch with multi-modal self-supervision," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[46] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, 2020, pp. 9912–9924.

[47] J. Li, P. Zhou, C. Xiong, and S. C. H. Hoi, "Prototypical contrastive learning of unsupervised representations," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021.

[48] L. Zhou, C. Xu, and J. J. Corso, "Towards automatic learning of procedures from web instructional videos," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, S. A. McIlraith and K. Q. Weinberger, Eds., AAAI Press, 2018, pp. 7590–7598.

[49] D. Zhukov, J. Alayrac, R. G. Cinbis, D. F. Fouhey, I. Laptev, and J. Sivic, "Cross-task weakly supervised learning from instructional videos," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 3537–3545.

[50] H. Kuehne, A. Iqbal, A. Richard, and J. Gall, "Mining youtube-a dataset for learning fine-grained action concepts from webly supervised video data," 2019.

[51] J. Carreira and A. Zisserman, "Quo vadis, action recognition? A new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 4724–4733.

[52] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," in *IJCV*, 2015.

[53] R. Sanabria *et al.*, "How2: a large-scale dataset for multimodal language understanding," in *Proceedings of the Workshop on Visually Grounded Interaction and Language (ViGIL)*, NeurIPS, 2018.

[54] E. Amrani, R. Ben-Ari, D. Rotman, and A. Bronstein, "Noise estimation using density estimation for self-supervised multimodal learning," in *AAAI*, 2021.

[55] J. Dong *et al.*, "Dual encoding for video retrieval by text," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 2021.

[56] V. Gabeur, C. Sun, K. Alahari, and C. Schmid, "Multi-modal transformer for video retrieval," 2020.

[57] J. Lei *et al.*, "Less is more: Clipbert for video-and-language learning via sparse sampling," 2021.

[58] H. Luo *et al.*, "Univilm: A unified video and language pre-training model for multimodal understanding and generation," vol. abs/2002.06353, 2020.

[59] M. Patrick *et al.*, "Support-set bottlenecks for video-text representation learning," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021.

[60] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 7463–7472.

[61] L. Zhu and Y. Yang, "Actbert: Learning global-local video-text representations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 8743–8752.

[62] A. Boggust *et al.*, "Grounding spoken words in unlabeled video," in *CVPRW*, 2019.

[63] Y. Liu, S. Albanie, A. Nagrani, and A. Zisserman, "Use what you have: Video retrieval using representations from collaborative experts," in *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, BMVA Press, 2019, p. 279.

[64] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, PMLR, vol. 119, PMLR, 2020, pp. 1597–1607.

[65] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 9726–9735.

[66] X. Yan, I. Misra, A. Gupta, D. Ghadiyaram, and D. Mahajan, "Clusterfit: Improving generalization of visual representations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 6508–6517.

[67] W. Van Gansbeke, S. Vandenhende, S. Georgoulis, M. Proesmans, and L. Van Gool, "Scan: Learning to classify images without labels," in *ECCV*, 2020.

[68] G. Ilharco, Y. Zhang, and J. Baldridge, "Large-scale representation learning from visually grounded untranscribed speech," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, Hong Kong, China: Association for Computational Linguistics, 2019, pp. 55–65.

[69] T. Chen and L. Li, "Intriguing properties of contrastive losses," vol. abs/2011.02803, 2020.

[70] L. Le, A. Patterson, and M. White, "Supervised autoencoders: Improving generalization performance with unsupervised regularizers," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 107–117.

[71] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 770–778.

[72] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, IEEE Computer Society, 2009, pp. 248–255.

[73] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 6546–6555.

[74] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2013.

[75] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[76] I. Misra and L. van der Maaten, "Self-supervised learning of pretext-invariant representations," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 6706–6716.

[77] P. Bojanowski *et al.*, "Weakly supervised action labeling in videos under ordering constraints," 2014.

[78] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," in *booktitle of machine learning research*, vol. 3, 2002, pp. 583–617.

[79] L. Hubert and P. Arabie, "Comparing partitions," in *booktitle of classification*, vol. 2, Springer, 1985, pp. 193–218.

[80] H. W. Kuhn, "The hungarian method for the assignment problem," in *Naval research logistics quarterly*, vol. 2, Wiley Online Library, 1955, pp. 83–97.

[81] B. Brattoli, J. Tighe, F. Zhdanov, P. Perona, and K. Chalupka, "Rethinking zero-shot video classification: End-to-end training for realistic applications," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 4612–4622.

[82] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," Tech. Rep., 2006.

[83] O. Russakovsky *et al.*, "Imagenet large scale visual recognition challenge," in *International booktitle of computer vision*, vol. 115, Springer, 2015, pp. 211–252.

[84] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, Eds., 2013, pp. 2292–2300.

[85] B. Chen *et al.*, "Multimodal clustering networks for self-supervised learning from unlabeled videos," *ArXiv preprint*, vol. abs/2104.12671, pp. 8012–8021, 2021.

[86] Y. Tang *et al.*, "COIN: A large-scale dataset for comprehensive instructional video analysis," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 1207–1216.

[87] N. Shvetsova *et al.*, "Everything at once–multi-modal fusion transformer for video retrieval," in *CVPR*, 2022.

[88] H. Akbari, S. Karaman, S. Bhargava, B. Chen, C. Vondrick, and S. Chang, "Multi-level multimodal common semantic space for image-phrase grounding," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 12 476–12 486.

[89] A. Arbelle *et al.*, "Detector-free weakly supervised grounding by separation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1801–1812.

[90] Z. Yang *et al.*, "Unitab: Unifying text and box outputs for grounded vision-language modeling," in *European Conference on Computer Vision*, Springer, 2022, pp. 521–539.

[91] L. H. Li *et al.*, "Grounded language-image pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 965–10 975.

[92] Z. Wang *et al.*, "Cris: Clip-driven referring image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11 686–11 695.

[93] Y. Zhong *et al.*, "Regionclip: Region-based language-image pretraining," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 793–16 803.

[94] R. Tan, B. Plummer, K. Saenko, H. Jin, and B. Russell, "Look at what i'm doing: Self-supervised spatial grounding of narrations in instructional videos," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[95] J. Shi, J. Xu, B. Gong, and C. Xu, "Not all frames are equal: Weakly-supervised video grounding with contextual similarity and visual clustering losses," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 444–10 452.

[96] T. Han, W. Xie, and A. Zisserman, "Temporal alignment networks for long-term video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2906–2916.

[97] Z. Chen, L. Ma, W. Luo, and K.-Y. K. Wong, "Weakly-supervised spatio-temporally grounding natural sentence in video," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 1884–1894.

[98] L. Zhou, N. Louis, and J. J. Corso, "Weakly-supervised video object grounding from text by loss weighting and object interaction," in *British Machine Vision Conference*, 2018.

[99] J. Weston, S. Bengio, and N. Usunier, "Wsabie: Scaling up to large vocabulary image annotation," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[100] A. Frome *et al.*, "Devise: A deep visual-semantic embedding model," *Advances in neural information processing systems*, vol. 26, 2013.

[101] R. Arandjelovic and A. Zisserman, "Look, listen and learn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 609–617.

[102] R. Arandjelović and A. Zisserman, "Objects that sound," in *ECCV*, 2018, pp. 435–451.

[103] B. Korbar, D. Tran, and L. Torresani, "Cooperative learning of audio and video models from self-supervised synchronization," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31, 2018, pp. 7774–7785.

[104] P. Morgado, N. Vasconcelos, and I. Misra, "Audio-visual instance discrimination with cross-modal agreement," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12 475–12 486.

[105] S. Ma, D. McDuff, and Y. Song, "Unpaired image-to-speech synthesis with multimodal information bottleneck," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7598–7607.

[106] Z. Li, R. Tao, E. Gavves, C. G. Snoek, and A. W. Smeulders, "Tracking by natural language specification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6495–6503.

[107] A. Sadhu, K. Chen, and R. Nevatia, "Video object grounding using semantic roles in language description," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 417–10 427.

[108] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Tubedetr: Spatio-temporal video grounding with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 442–16 453.

[109] *Mexaction2: Action detection and localization dataset*, http://mexculture.cnam.fr/xwiki/bin/view/Datasets/Mex+action+dataset.

[110] Y.-G. Jiang *et al.*, *Thumos challenge: Action recognition with a large number of classes*, http://crcv.ucf.edu/THUMOS14/, 2014.

[111] F. Heilbron, V Escorcia, B Ghanem, and J Niebles, "A large-scale video benchmark for human activity understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, 2015. 961*, vol. 970, 2019.

[112] M. Soldan *et al.*, "Mad: A scalable dataset for language grounding in videos from movie audio descriptions," *arXiv preprint arXiv:2112.00431*, 2021.

[113] M. Regneri, M. Rohrbach, D. Wetzel, S. Thater, B. Schiele, and M. Pinkal, "Grounding action descriptions in videos," *Transactions of the Association for Computational Linguistics*, vol. 1, pp. 25–36, 2013.

[114] J. Gao, C. Sun, Z. Yang, and R. Nevatia, "Tall: Temporal activity localization via language query," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5267–5275.

[115] J. Chen, X. Chen, L. Ma, Z. Jie, and T.-S. Chua, "Temporally grounding natural sentence in video," in *Proceedings of the 2018 conference on empirical methods in natural language processing*, 2018, pp. 162–171.

[116] Y. Yuan, L. Ma, J. Wang, W. Liu, and W. Zhu, "Semantic conditioned dynamic modulation for temporal sentence grounding in videos," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[117] Z. Zhang, Z. Lin, Z. Zhao, and Z. Xiao, "Cross-modal interaction networks for query-based moment retrieval in videos," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 655–664.

[118] H. Xu, K. He, B. A. Plummer, L. Sigal, S. Sclaroff, and K. Saenko, "Multilevel language and vision integration for text-to-clip retrieval," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9062–9069.

[119] J. Wang, L. Ma, and W. Jiang, "Temporally grounding language queries in videos by contextual boundary-aware prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 12 168–12 175.

[120] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1049–1058.

[121] V. Escorcia, F. Caba Heilbron, J. C. Niebles, and B. Ghanem, "Daps: Deep action proposals for action understanding," in *European conference on computer vision*, Springer, 2016, pp. 768–784.

[122] C. Rodriguez, E. Marrese-Taylor, F. S. Saleh, H. Li, and S. Gould, "Proposal-free temporal moment localization of a natural-language query in video using guided attention," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 2464–2473.

[123] Y. Yuan, T. Mei, and W. Zhu, "To find where you talk: Temporal sentence localization in video with attention based location regression," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 9159–9166.

[124] M. Hahn, A. Kadav, J. M. Rehg, and H. P. Graf, "Tripping through time: Efficient localization of activities in videos," *arXiv preprint arXiv:1904.09936*, 2019.

[125] W. Wang, Y. Huang, and L. Wang, "Language-driven temporal activity localization: A semantic matching reinforcement learning model," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 334–343.

[126] R. Zeng, H. Xu, W. Huang, P. Chen, M. Tan, and C. Gan, "Dense regression network for video grounding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 287–10 296.

[127] Y. Zhao, Z. Zhao, Z. Zhang, and Z. Lin, "Cascaded prediction network via segment tree for temporal video grounding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4197–4206.

[128] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *ECCV*, 2018, pp. 305–321.

[129] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, "Stacked cross attention for image-text matching," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 201–216.

[130] S. Abnar and W. Zuidema, "Quantifying attention flow in transformers," *arXiv preprint arXiv:2005.00928*, 2020.

[131] C. Gu *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*, 2018, pp. 6047–6056.

[132] V. Kalogeiton, P. Weinzaepfel, V. Ferrari, and C. Schmid, "Action tubelet detector for spatio-temporal action localization," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4405–4413.

[133] S. Li, Y. Du, A. Torralba, J. Sivic, and B. Russell, "Weakly supervised human-object interaction detection in video via contrastive spatiotemporal regions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1845–1855.

[134] P. Weinzaepfel, X. Martin, and C. Schmid, "Human action localization with sparse spatial supervision," *arXiv preprint arXiv:1605.05197*, 2016.

[135] H. Luo *et al.*, "Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning," *Neurocomputing*, vol. 508, pp. 293–304, 2022.

[136] P. Mettes and C. G. Snoek, "Spatial-aware object embeddings for zero-shot localization and classification of actions," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4443–4452.

[137] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 3733–3742.

[138] M. Ye, X. Zhang, P. C. Yuen, and S. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 6210–6219.

[139] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *ArXiv preprint*, 2018.

[140] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *ECCV*, 2020.

[141] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[142] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *ArXiv preprint*, 2020.

[143] D. Pathak, R. B. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning features by watching objects move," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, IEEE Computer Society, 2017, pp. 6024–6033.

[144] A. Owens, J. Wu, J. H. McDermott, W. T. Freeman, and A. Torralba, "Ambient sound provides supervision for visual learning," in *European conference on computer vision*, Springer, 2016.

[145] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: Unsupervised learning using temporal order verification," in *ECCV*, Springer, 2016, pp. 527–544.

[146] X. Wang, K. He, and A. Gupta, "Transitive invariance for self-supervised visual representation learning," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 1338–1347.

[147] H. Lee, J. Huang, M. Singh, and M. Yang, "Unsupervised representation learning by sorting sequences," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 667–676.

[148] U. Buchler, B. Brattoli, and B. Ommer, "Improving spatiotemporal self-supervision by deep reinforcement learning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[149] D. Wei, J. J. Lim, A. Zisserman, and W. T. Freeman, "Learning and using the arrow of time," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 8052–8060.

[150] L. Jing and Y. Tian, "Self-supervised spatiotemporal feature learning by video geometric transformations," in *arXiv preprint*, 2018.

[151] N. Sayed, B. Brattoli, and B. Ommer, "Cross and learn: Cross-modal self-supervision," in *German Conference on Pattern Recognition*, Springer, 2018.

[152] S. Purushwalkam and A. Gupta, "Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[153] A. Jabri, A. Owens, and A. A. Efros, "Space-time correspondence as a contrastive random walk," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[154] D. Gordon, K. Ehsani, D. Fox, and A. Farhadi, "Watching the world go by: Representation learning from unlabeled videos," *ArXiv preprint*, 2020.

[155] J. Wang *et al.*, "Removing the background by adding the background: Towards background robust self-supervised video representation learning," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[156] J. Choi, C. Gao, J. C. E. Messou, and J. Huang, "Why can't I dance in the mall? learning to mitigate scene bias in action recognition," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 851–863.

[157] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 618–626.

[158] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. V. Gool, M. H. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *2016*

IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 724–732.

[159] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 15 509–15 519.

[160] O. J. Hénaff, "Data-efficient image recognition with contrastive predictive coding," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 4182–4192.

[161] C. Zhuang, A. L. Zhai, and D. Yamins, "Local aggregation for unsupervised learning of visual embeddings," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 6001–6011.

[162] S. Benaim *et al.*, "Speednet: Learning the speediness in videos," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 9919–9928.

[163] P. Agrawal, J. Carreira, and J. Malik, "Learning to see by moving," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, IEEE Computer Society, 2015, pp. 37–45.

[164] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, IEEE Computer Society, 2015, pp. 2794–2802.

[165] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy, "Tracking emerges by colorizing videos," in *ECCV*, 2018, pp. 391–408.

[166] R. Qian *et al.*, "Spatiotemporal contrastive video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6964–6974.

[167] K. Ranasinghe, M. Naseer, S. Khan, F. S. Khan, and M. S. Ryoo, "Self-supervised video transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2874–2884.

[168] K. Hu, J. Shao, Y. Liu, B. Raj, M. Savvides, and Z. Shen, "Contrast and order representations for video self-supervised learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7939–7949.

[169] Z. Qing *et al.*, "Learning from untrimmed videos: Self-supervised video representation learning with hierarchical consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 821–13 831.

[170] J. Xu and X. Wang, "Rethinking self-supervised correspondence learning: A video frame-level similarity perspective," *ArXiv preprint*, 2021.

[171] D. Dwibedi, Y. Aytar, J. Tompson, P. Sermanet, and A. Zisserman, "Temporal cycle-consistency learning," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 1801–1810.

[172] O. J. Hénaff, S. Koppula, J.-B. Alayrac, A. v. d. Oord, O. Vinyals, and J. Carreira, "Efficient visual pretraining with contrastive detection," *ArXiv preprint*, 2021.

[173] S. Mo, H. Kang, K. Sohn, C.-L. Li, and J. Shin, "Object-aware contrastive learning for debiased scene representation," *ArXiv preprint*, 2021.

[174] J. Xie, X. Zhan, Z. Liu, Y. S. Ong, and C. C. Loy, "Unsupervised object-level representation learning from scene images," *ArXiv preprint*, 2021.

[175] R. R. Selvaraju, K. Desai, J. Johnson, and N. Naik, "Casting your model: Learning to localize improves self-supervised representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[176] S. Ding *et al.*, "Motion-aware contrastive video representation learning via foreground-background merging," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9716–9726.

[177] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, IEEE Computer Society, 2015, pp. 815–823.

[178] D. T. Nguyen *et al.*, "Deepusps: Deep robust unsupervised saliency prediction via self-supervision," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 204–214.

[179] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE international conference on image processing (ICIP)*, IEEE, 2016.

[180] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, "Vggsound: A large-scale audio-visual dataset," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, IEEE, 2020, pp. 721–725.

[181] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *IJCV*, 2009.

[182] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *ECCV*, 2014.

[183] K. Desai and J. Johnson, "Virtex: Learning visual representations from textual annotations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.

[184] S. Guo *et al.*, "Cross-architecture self-supervised video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 270–19 279.

[185] J. Wang, J. Jiao, and Y.-H. Liu, "Self-supervised video representation learning by pace prediction," in *European Conference on Computer Vision*, Springer, 2020, pp. 504–521.

[186] J. Wang, J. Jiao, L. Bao, S. He, W. Liu, and Y.-H. Liu, "Self-supervised video representation learning by uncovering spatio-temporal statistics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[187] T. Pan, Y. Song, T. Yang, W. Jiang, and W. Liu, "Videomoco: Contrastive video representation learning with temporally adversarial examples," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 205–11 214.

[188] T. Han, W. Xie, and A. Zisserman, "Self-supervised co-training for video representation learning," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020.

[189] D. Huang *et al.*, "Ascnet: Self-supervised video representation learning with appearance-speed consistency," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[190] I. Dave, R. Gupta, M. N. Rizve, and M. Shah, "Tclr: Temporal contrastive learning for video representation," *arXiv preprint arXiv:2101.07974*, 2021.

[191] D. Xu, J. Xiao, Z. Zhao, J. Shao, D. Xie, and Y. Zhuang, "Self-supervised spatiotemporal learning via video clip order prediction," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, Computer Vision Foundation / IEEE, 2019, pp. 10 334–10 343.

[192] K. Y. Xiao, L. Engstrom, A. Ilyas, and A. Madry, "Noise or signal: The role of image backgrounds in object recognition," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, OpenReview.net, 2021.

[193] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black, "Towards understanding action recognition," in *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, IEEE Computer Society, 2013, pp. 3192–3199.

[194] Y. Li, Y. Li, and N. Vasconcelos, "Resound: Towards action recognition without representation bias," *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[195] J. Choi, C. Gao, J. C. E. Messou, and J.-B. Huang, "Why can't i dance in the mall? learning to mitigate scene bias in action recognition," in *Advances in Neural Information Processing Systems*, 2019.

[196] K. Maninis, S. Caelles, J. Pont-Tuset, and L. V. Gool, "Deep extreme cut: From extreme points to object segmentation," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, IEEE Computer Society, 2018, pp. 616–625.

[197] T. H. Nguyen, K. Cho, and R. Grishman, "Joint event extraction via recurrent neural networks," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, California: Association for Computational Linguistics, 2016, pp. 300–309.

[198] L. Sha, F. Qian, B. Chang, and Z. Sui, "Jointly extracting event triggers and arguments by dependency-bridge RNN and tensor-based argument interaction," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, S. A. McIlraith and K. Q. Weinberger, Eds., AAAI Press, 2018, pp. 5916–5923.

[199] J. Liu, Y. Chen, K. Liu, and J. Zhao, "Neural cross-lingual event detection with minimal parallel resources," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China: Association for Computational Linguistics, 2019, pp. 738–748.

[200] J. Liu, Y. Chen, K. Liu, W. Bi, and X. Liu, "Event extraction as machine reading comprehension," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online: Association for Computational Linguistics, 2020, pp. 1641–1651.

[201] X. Wang *et al.*, "HMEAE: Hierarchical modular event argument extraction," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China: Association for Computational Linguistics, 2019, pp. 5777–5783.

[202] X. Wang *et al.*, "Neural Gibbs Sampling for Joint Event Argument Extraction," in *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, Suzhou, China: Association for Computational Linguistics, 2020, pp. 169–180.

[203] M. Yatskar, L. S. Zettlemoyer, and A. Farhadi, "Situation recognition: Visual semantic role labeling for image understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, IEEE Computer Society, 2016, pp. 5534–5542.

[204] R. Li, M. Tapaswi, R. Liao, J. Jia, R. Urtasun, and S. Fidler, "Situation recognition with graph neural networks," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 4183–4192.

[205] A. Mallya and S. Lazebnik, "Recurrent models for situation recognition," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2017, pp. 455–463.

[206] S. Pratt, M. Yatskar, L. Weihs, A. Farhadi, and A. Kembhavi, "Grounded situation recognition," in *European Conference on Computer Vision*, Springer, 2020, pp. 314–332.

[207] T. Zhang *et al.*, "Improving event extraction via multimodal integration," in *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, 2017, pp. 270–278.

[208] M. Tong *et al.*, "Image enhanced event detection in news articles," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, AAAI Press, 2020, pp. 9040–9047.

[209] M. Li *et al.*, "Cross-media structured common space for multimedia event extraction," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, 2020, pp. 2557–2568.

[210] A. Pardo, H. Alwassel, F. Caba, A. Thabet, and B. Ghanem, "Refineloc: Iterative refinement for weakly-supervised action localization," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 3319–3328.

[211] Y. Huang, Y. Sugano, and Y. Sato, "Improving action segmentation via graph-based temporal reasoning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 14 021–14 031.

[212] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "BMN: boundary-matching network for temporal action proposal generation," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 3888–3897.

[213] Z. Zhang, Z. Zhao, Z. Lin, X. He, *et al.*, "Counterfactual contrastive learning for weakly-supervised vision-language grounding," *Advances in Neural Information Processing Systems*, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., pp. 18 123–18 134, 2020.

[214] T. Gupta, A. Vahdat, G. Chechik, X. Yang, J. Kautz, and D. Hoiem, "Contrastive learning for weakly supervised phrase grounding," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, Springer, 2020, pp. 752–768.

[215] H. Yang, Y. Chen, K. Liu, Y. Xiao, and J. Zhao, "DCFEE: A document-level Chinese financial event extraction system based on automatically labeled training data," in *Proceedings of ACL 2018, System Demonstrations*, Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 50–55.

[216] S. Li, H. Ji, and J. Han, "Document-level event argument extraction by conditional generation," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, 2021, pp. 894–908.

[217] Y. Zeng *et al.*, "Scale up event extraction learning via automatic training data generation," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, S. A. McIlraith and K. Q. Weinberger, Eds., AAAI Press, 2018, pp. 6045–6052.

[218] D. Ahn, "The stages of event extraction," in *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, Sydney, Australia: Association for Computational Linguistics, 2006, pp. 1–8.

[219] H. Ji and R. Grishman, "Refining event extraction through cross-document inference," in *Proceedings of ACL-08: HLT*, Columbus, Ohio: Association for Computational Linguistics, 2008, pp. 254–262.

[220] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Beijing, China: Association for Computational Linguistics, 2015, pp. 167–176.

[221] Z. Zhang and H. Ji, "Abstract Meaning Representation guided graph encoding and decoding for joint information extraction," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, 2021, pp. 39–49.

[222] Y. Lin, H. Ji, F. Huang, and L. Wu, "A joint end-to-end neural model for information extraction with global features," in *Proc. The 58th Annual Meeting of the Association for Computational Linguistics (ACL2020)*, 2020.

[223] L. Huang, H. Ji, K. Cho, I. Dagan, S. Riedel, and C. Voss, "Zero-shot transfer learning for event extraction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia: Association for Computational Linguistics, 2018, pp. 2160–2170.

[224] Y. Chen, S. Liu, X. Zhang, K. Liu, and J. Zhao, "Automatically labeled data generation for large scale event extraction," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 409–419.

[225] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The Berkeley FrameNet project," in *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*, 1998.

[226] M. Suhail and L. Sigal, "Mixture-kernel graph attention network for situation recognition," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, IEEE, 2019, pp. 10 362–10 371.

[227] T. Cooray, N. Cheung, and W. Lu, "Attention-based context aware reasoning for situation recognition," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, IEEE, 2020, pp. 4735–4744.

[228] A. Sadhu, T. Gupta, M. Yatskar, R. Nevatia, and A. Kembhavi, "Visual semantic role labeling for video understanding," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[229] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "Bsn: Boundary sensitive network for temporal action proposal generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[230] H. Su, W. Gan, W. Wu, Y. Qiao, and J. Yan, "Bsn++: Complementary boundary regressor with scale-balanced relation modeling for temporal action proposal generation," 2021.

[231] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," *ArXiv preprint*, vol. abs/1602.02410, 2016.

[232] X. Lin, G. Bertasius, J. Wang, S.-F. Chang, D. Parikh, and L. Torresani, "Vx2text: End-to-end learning of video-based text generation from multimodal inputs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 7005–7015.

[233] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[234] Y. Lin, H. Ji, F. Huang, and L. Wu, "A joint neural model for information extraction with global features," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online: Association for Computational Linguistics, 2020, pp. 7999–8009.

[235] A. Karpathy, A. Joulin, and F. Li, "Deep fragment embeddings for bidirectional image sentence mapping," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., 2014, pp. 1889–1897.

[236] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., 2015, pp. 91–99.

[237] R. Krishna *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, vol. 123, no. 1, pp. 32–73, 2017.

[238] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45.

[239] X. Lin, F. Petroni, G. Bertasius, M. Rohrbach, S.-F. Chang, and L. Torresani, "Learning to recognize procedural activities with distant supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 853–13 863.

[240] A. Rouditchenko *et al.*, "Cascaded multilingual audio-visual learning from videos," *arXiv preprint arXiv:2111.04823*, 2021.