Identity-Text Video Corpus Grounding

Bin Huang[1], **Xin Wang**[*1, 2], **Hong Chen**[1], **Houlun Chen**[1], **Yaofei Wu**[3], **Wenwu Zhu**[*1, 2]

[1]Department of Computer Science and Technology, Tsinghua University
[2]BNRIST, Tsinghua University
[3]Beijing University of Technology
{huangb23, h-chen20, chenhl23}@mails.tsinghua.edu.cn
{xin_wang,wwzhu}@tsinghua.edu.cn, 23027313@emails.bjut.edu.cn

Abstract

Video corpus grounding (VCG), which aims to retrieve relevant video moments from a video corpus, has attracted significant attention in the multimedia research community. However, the existing VCG setting primarily focuses on matching textual descriptions with videos and ignores the distinct visual identities in the videos, thus resulting in inaccurate understanding of video content and deteriorated retrieval performances. To address this limitation, we introduce a novel task, Identity-Text Video Corpus Grounding (ITVCG), which simultaneously utilize textual descriptions and visual identities as queries. As such, ITVCG benefits in enabling more accurate video corpus grounding with visual identities, as well as providing users with more flexible options to locate relevant frames based on either *textual descriptions* or *textual descriptions and visual identities*. To conduct evaluations regarding the novel ITVCG task, we propose the TVR-IT dataset, comprising 463 identity images from 6 TV shows, with 68,840 out of 72,840 queries containing at least one identity image. Furthermore, we propose Video-Locator, the first model designed for the ITVCG task. Our proposed Video-Locator integrates video-identity-text alignment and multi-modal fine-grained fusion components, facilitating a video large language model (Video LLM) to jointly understand textual descriptions, visual identities, as well as videos. Experimental results demonstrate the effectiveness of the proposed Video-Locator model and highlight the importance of identity-generalization capability for ITVCG. Our project page is at https://github.com/huangb23/Identity-Text-Video-Corpus-Grounding

Introduction

In recent years, the task of Video Corpus Grounding (VCG)(Escorcia et al. 2019) has garnered significant attention within the multimedia research community, which aims to retrieve the most relevant video moments from an extensive video corpus. This task holds immense potential for a variety of downstream applications, including but not limited to video editing, recommendation, and content creation.

Despite the advancements in video corpus grounding, existing works primarily focus on retrieving video moments solely based on textual descriptions as input, failing to distinguish the visual identities in the videos. For example, a teenager may desire to locate a video moment that involves his favorite movie star dancing, while existing works can only locate video moments of dancing, failing to guarantee the retrieved videos contain the favorite movie star.

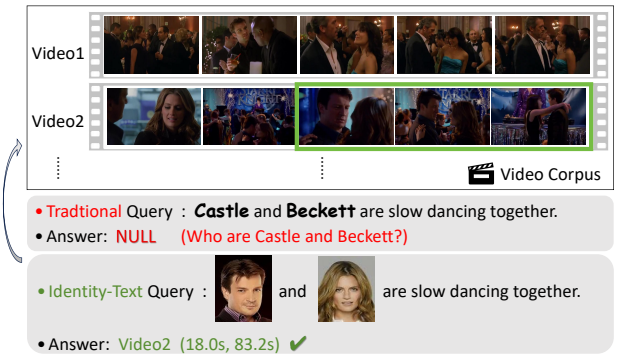

Figure 1: Traditional Video Corpus Grounding fails to identify identities within videos. Our proposed Identity-Text Video Corpus Grounding task incorporates visual identities into the query, enabling more precise and flexible video grounding.

To tackle the problem, in this paper we propose a novel task, Identity-Text Video Corpus Grounding (ITVCG), which aims to locate the most relevant segments from video corpus given both textual descriptions (texts) and visual identities (images) as shown in Figure 1. Identity-Text Video Corpus Grounding has a broader range of applications than traditional Video Corpus Grounding, benefiting in more accurate video corpus grounding with visual identities, as well as providing users with more flexible options to locate relevant frames based on either *textual descriptions* or *textual descriptions and visual identities*.

However, despite the significance of the novel ITVCG task, there are currently no existing works in literature, lacking both datasets and models. To solve this issue, we propose the TVR-IT dataset and the Video-Locator model for Identity-Text Video Corpus Grounding, which we believe will have a broad impact on following works for Video Corpus Grounding. Specifically, we construct the TVR-IT

*Corresponding Authors.

dataset based on the TVR(Lei et al. 2020) dataset. We gather 463 identity images from 6 TV shows. Among the final 72,840 queries, 68,840 ($\sim 94.5\%$) contain at least one identity image. Furthermore, we provide two settings, i.e., the identity-seen and identity-unseen dataset splits, to better evaluate the generalization ability of the model. Besides the dataset, we further propose Video-Locator, which comprises two primary components: i) the Video-Identity-Text Alignment module, which separately processes videos with visual identities and queries with identity images, aligning their representations using contrastive learning to enable rapid retrieval of video(s) containing the ground truth moment within the video corpus; and ii) the Multi-Modal Fine-Grained Fusion module, which integrates the representations of the matched videos and queries to generate precise start and end timestamps for the relevant video moments. Additionally, to enhance the model's generalization capability, we utilize a large language model (LLM) as the backbone and conduct extensive pre-training with visual-text data. We conduct experiments to show the effectiveness of the proposed Video-Locator model, as well as indicating the importance of identity-generalization for ITVCG task.

To summarize, our contributions are listed as follows,

- To the best of our knowledge, we are the first to introduce the novel and important task, Identity-Text Video Corpus Grounding (ITVCG).
- We propose the TVR-IT dataset that involves 463 identities and 72,840 joint identity-text queries, with identity-seen and identity-unseen generalization splits for out-of-distribution (OOD) test in the ITVCG task.
- We propose the Video-Locator model for ITVCG, comprising a Video-Identity-Text Alignment module and a Multi-Modal Fine-Grained Fusion module, enabling accurate video corpus grounding.
- We conduct extensive experiments to demonstrate the effectiveness of the Video-Locator model, and present our discoveries on the importance of identity-generalization for ITVCG.

Related Works

Video Corpus Grounding Datasets

Video corpus grounding(Escorcia et al. 2019), also known as video corpus moment retrieval, involves identifying video segments from a large collection that match a given query. This task is typically divided into two subtasks: video retrieval (VR), where the model searches for videos likely to contain the target segment within a large corpus, also referred to as partially relevant video retrieval(Dong et al. 2022), and single video grounding (SVG), which focuses on identifying the time segment within an individual video. (Chen et al. 2023a; Feng et al. 2023; Chen et al. 2023b; Wang et al. 2023; Wang, Lan, and Zhu 2022; Yuan et al. 2021)

Initially, video corpus grounding was approached by extending SVG datasets(Yu et al. 2019; Anne Hendricks et al. 2017; Gao et al. 2017; Chen et al. 2024a) directly for the VCG task. TVR(Lei et al. 2020) constructed the dataset from TV shows as they typically involve rich social interactions between identities, approximately 66% of queries contain two or more identity names. However, it is hard to accurately associate names with identity faces in videos. Additionally, we may wish to search for video moments featuring a specific movie star without specifying the character they portray. To resolve these issues, we propose a novel task: Identity-Text Video Corpus Grounding, which incorporates identity images into the query, and we collect relevant identity images to construct our TVR-IT dataset.

Video Corpus Grounding Models

Many researchers have developed effective models for the VCG task. For example, XML(Lei et al. 2020) utilizes a two-tower model to jointly learn video retrieval and moment localization objectives. ReLoCLNet(Zhang et al. 2021) introduces contrastive learning between query-video and query-frame pairs to encourage alignment at different granularities. SQuiDNet(Yoon et al. 2022) employs causal inference to prevent the model from learning incorrect retrieval biases. HERO(Li et al. 2020) undergoes video-language pre-training and is then fine-tuned on the TVR dataset for video corpus grounding. EventFormer(Hou et al. 2024) aggregates frames to form events and explicitly leverages event-level information interaction. These models have all achieved strong performance on the VCG task. However, they are not directly applicable to our ITVCG task, as they cannot handle joint identity-text queries.

In this paper, we introduce Video-Locator, the first model specifically designed for the ITVCG task, which enables a Video LLM to jointly understand and process text, identity images, and videos.

Identity-Text Video Corpus Grounding

In this section, we first define the Identity-Text Video Corpus Grounding (ITVCG) task, followed by an introduction to the TVR-IT dataset that we constructed for the task.

Task Formulation

We first revisit the task of traditional Video Corpus Grounding (VCG). We denote a video corpus that contains many videos as $\mathcal{V} = \{V_1, V_2, ...\}$, where V_i indicates the i^{th} video. Given a query $Q = [q_i]_{i=1}^{n_q}$ where q_i represents a word, the objective of VCG is to retrieve the temporal moment (τ^s, τ^e) in V^* that semantically aligns with Q. Here, V^* is a video containing the ground truth moment within the video corpus $\mathcal{V}$. The notation (τ^s, τ^e) represents the start and end timestamps of the moment.

In traditional VCG, the query Q only involves textual words, which cannot fulfill the personalized retrieval requirements (e.g., a boy's favorite movie star) with visual details. In contrast, in our proposed ITVCG, we incorporate the identity image together into the query, e.g. *"$Image_1$ is dancing with $Image_2$ in the gym"*. ITVCG has a broader range of applications than traditional VCG, where it enables users to search video segments through not only text descriptions but also visual images, which is more accurate and flexible. In the following section, we introduce the TVR-IT dataset that we construct for the task.

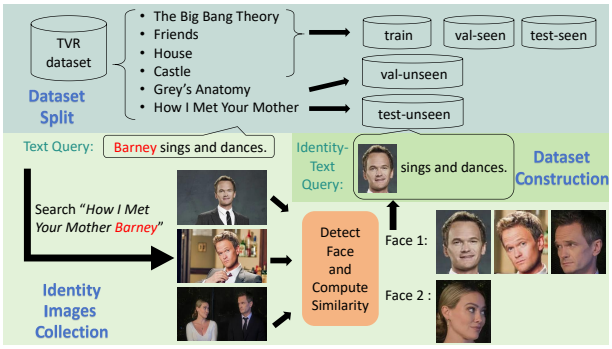

Figure 2: An illustration of the construction process for the TVR-IT dataset, which involves three stages: identity image collection, dataset construction, and dataset split.

TVR-IT dataset

We construct our TVR-IT dataset based on the TVR(Lei et al. 2020) dataset, which encompasses 109K queries collected from 21.8K videos across 6 TV shows of various genres. Based on the original textual queries and videos in TVR dataset, we can conduct traditional VCG. However, for the ITVCG task, we additionally require the identity images. Next, we will elaborate on how we collect the identity images and how we construct the TVR-IT dataset based on the identity images. An illustration for the dataset construction is shown in Figure 2.

Identity images collection We identify identity names appearing at least 3 times within each TV show, conduct Google searches using the respective TV show name appended with the identity name (e.g., *"How I Met Your Mother Barney"*), and download the top 3 ranking images as candidate images. These images typically include the identity's face, but may also contain faces of other individuals, necessitating further processing.

We utilize a pre-trained face model, Antelopev2[1], to detect faces in the images and compute face embeddings for each detected face. To determine whether two faces correspond to the same identity, we compute the cosine similarity between their respective face embeddings, denoted as f_1 and f_2, as follows:

$$\cos(f_1, f_2) = \frac{f_1^T f_2}{||f_1|| \cdot ||f_2||}. \quad (1)$$

If the cosine similarity between two face embeddings surpassed a threshold $\theta = 0.4$, i.e., $\cos(f_1, f_2) > \theta$, we consider them to represent the same face. When only one face appears in all three images, we confirm that it is the face of the desired identity. Otherwise, we resort to manual verification. We ultimately collect 463 identity images from 6 TV shows.

Dataset construction After we collected the identity images, we construct our TVR-IT dataset as follows. For each textual query in the original TVR dataset, we will replace the identity names with the corresponding identity images. For example, the original textual query *"Barney sings and dances"* will be converted into a multi-modal query *"$Image_1$ sings and dances"*. Additionally, note that the original TVR dataset includes subtitles corresponding to the videos, and previous works also utilize the subtitles to match the queries which is a text-to-text retrieval auxiliary task to help video corpus grounding. However, in our work, we expect the model to fully understand the cross-modality relations among text, identity images and videos, and therefore we remove the subtitles. This also makes our work can be applied to broader scenarios where the videos do not incorporate subtitles. Finally, we obtain 72,840 queries, 68,840 ($\sim$94.5%) contained at least one identity image.

[1]https://github.com/deepinsight/insightface

Table 1: The information of TVR-IT dataset. "# ID" represents the number of identities with corresponding images. BBT=*The Big Bang Theory*, Grey=*Grey's Anatomy*, HIMYM=*How I Met Your Mother*.

Split	# Videos	# Queries	Shows	# ID
train	15060	56735	BBT,Friends, House,Castle	374
val-seen	555	2084		
test-seen	1371	5142		
val-unseen	1257	4157	Grey	52
test-unseen	1371	4722	HIMYM	37

Dataset split To better assess the generalization capability of the model, we provide **(identity-)seen** and **(identity-)unseen** splits for both the validation and test sets. Specifically, four TV shows are utilized to construct the train, val-seen, and test-seen sets, while one TV show is used for the val-unseen set, and the remaining one TV show is used for the test-unseen set. Videos and queries in the seen split do not appear in the training set, they are just from the same TV show. The identity-seen split helps us to evaluate whether the model can generalize to different videos and text queries with the same identities as training. In contrast, the identity-unseen split is more challenging, the videos, text queries, and identities are all different from the training dataset, which requires a model with stronger generalization ability.

Video-Locator

In this section, we present Video-Locator, a novel model that leverages the capabilities of a Large Language Model (LLM) for the ITVCG task. We will first introduce the preliminaries about LLM. (Chen et al. 2024b) Subsequently, we will provide the details on how the Video-Locator is built on LLM for the ITVCG task. Lastly, we will discuss the inference process of Video-Locator.

Preliminaries

The LLM typically comprises an embedding layer $Embed(\cdot)$, L layers of decoder $[Decoder_l(\cdot)]_{l=1}^{L}$, and an output head $Output(\cdot)$. Given a query $Q = [q_i]_{i=1}^{n_q}$, the LLM first processes the input query Q by passing it through

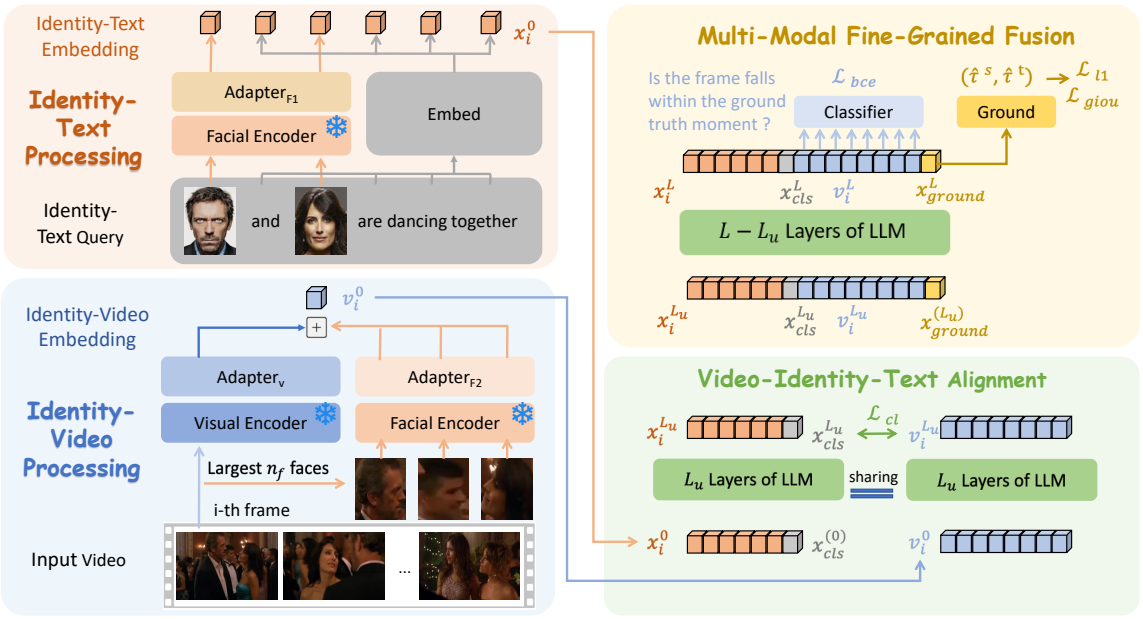

Figure 3: The framework of the Video-Locator model. We introduce separate facial adapters and a visual Adapter to enable the LLM to comprehend facial and visual features. Subsequently, Identity-Text and Identity-Video features are aligned through the Video-Identity-Text Alignment module. The Multi-Modal Fine-Grained Fusion module integrates representations of matched videos and queries to generate timestamps.

the embedding layer, resulting in a sequence of vector representations:

$$[x_i^0]_{i=1}^{n_q} = Embed([q_i]_{i=1}^{n_q}). \quad (2)$$

This sequence is passed through the L decoder layers sequentially, which applies self-attention and feed-forward neural networks to generate a new sequence of hidden states:

$$[x_i^l]_{i=1}^{n_q} = Decoder_l([x_i^{l-1}]_{i=1}^{n_q}), \ \ l = 1, 2, ..., L. \quad (3)$$

Finally, the output head takes the hidden states produced by the last decoder layer and computes a probability distribution over the possible next tokens:

$$p_{n_q+1} = Output(x_{n_q}^L). \quad (4)$$

The token with the highest probability is selected as the LLM's prediction for the next token in the sequence.

To adapt the LLM for the ITVCG task, we employ a two-stage training approach. In the first stage, we introduce a visual adapter to enable LLM to comprehend visual features of video frames, and pretrain it using a large number of image-text pairs. In the second stage, we build our Video-Locator model based on the LLM as shown in Figure 3, which includes:

- Utilize the pretrained Visual Adapter to enable LLM to comprehend visual features.
- Introduce the Facial Adapter to enable LLM to understand identity images.
- Utilize half of LLM decoders (close to the input) as a Video-Identity-Text Alignment module, facilitating rapid video retrieval.
- Leverage half of the LLM decoders (close to the ouput) as a Multi-Modal Fine-Grained Fusion module, integrating representations of matched videos and queries to generate timestamps.

Visual Adapter

To enable the LLM to comprehend visual data, it is necessary to introduce a visual adapter, which serves to map the features extracted by a frozen CLIP(Radford et al. 2021) ViT-L/14 model into the feature space of the LLM.

For an image, a 768-dimensional embedding v^{clip} is obtained through CLIP. Subsequently, we apply a two-layer MLP, denoted as $Adapter_V(\cdot)$, to project the feature into the embedding space of the LLM, as follows:

$$x_0^0 = Adapter_V(v^{clip}) \in R^d \quad (5)$$

where d is the hidden dimension of LLM. The resulting vector, x_0^0, contains the visual information that can be understood by the LLM.

Following (Liu et al. 2023; Huang et al. 2024), it is necessary to pretrain the visual adapter using a large number of image-text pairs. We utilize the LCS-558K dataset curated by LLaVA(Liu et al. 2023). We want the model to generate corresponding text given an image. Specifically, for a text-image pair, we obtain its image embedding x_0^0 from the initial visual embedding v^{clip} through Eq.(5), and obtain the embeddings $[x_i^0]_{i=1}^{n_q}$ of the text $Q = [q_i]_{i=1}^{n_q}$ through Eq.(2). We then place the image embedding before the text embeddings and get a sequence $[x_i^0]_{i=0}^{n_q}$, and then train on the se-

quence using the next token prediction task, as follows:

$$[x_i^l]_{i=0}^{n_q} = Decoder_l([x_i^{l-1}]_{i=0}^{n_q}), \ \ l = 1, 2, ..., L, \quad (6)$$

$$[p_i]_{i=1}^{n_q} = [Output(x_{i-1}^L)]_{i=1}^{n_q}, \quad (7)$$

$$\mathcal{L}_{\text{pre}} = -\frac{1}{n_q}\sum_{i=1}^{n_q}\log(p_{i,id(q_i)}), \quad (8)$$

where $id(q_i)$ is the index of the word q_i in the vocabulary, $p_{i,id(q_i)}$ is the predicted probability of generating the word q_i at the i-th position. During the pre-training stage, only the visual adapter is trained, while the LLM is kept frozen.

Next, we will elaborate on how we develop the Video-Locater based on the LLM and Visual Adapter for the ITVCG task.

Facial Adapter

In the ITVCG task, the query contains identity images, and videos may also contain numerous faces. We employ the pre-trained face model, Antelopev2, to compute the embedding for each face. Similarly, we introduce facial adapters on both the query and video sides, enabling our Video Locator to recognize faces.

Specifically, for a query $Q = [q_i]_{i=1}^{n_q}$, where q_i could be a word or a identity image, we introduce $Adapter_{F1}(\cdot)$ to calculate the vector representation of Q as follows,

$$x_i^0 = \begin{cases} Embed(q_i) & \text{if } q_i \text{ is a word} \\ Adapter_{F1}(Antelopev2(q_i)) & \text{if } q_i \text{ is an image} \end{cases}, \quad (9)$$

so we can process this query embedding sequence $[x_i^0]_{i=1}^{n_q}$.

For a given video V, we uniformly sample n_v frames. For each frame, we utilize Antelopev2 to detect faces and compute face embeddings for each detected face. We retain only the n_f faces with the largest areas in each frame. If there are fewer than n_f faces, the corresponding embeddings are filled with zero vectors. Both n_v and n_f are predefined constants. This process yields a sequence of face embeddings for the video $[(f_{i,1}, f_{i,2}, ..., f_{i,n_f})]_{i=1}^{n_v}$. We introduce $Adapter_{F2}(\cdot)$ to map the face embeddings from the video side to the LLM space. Additionally, we obtain a sequence of visual features $[v_i^{clip}]_{i=1}^{n_v}$ from CLIP, which are then mapped using the pre-trained visual adapter and added to the face features to obtain the vector representation of V as follows:

$$v_i^0 = Adapter_V(v_i^{clip}) + \sum_{j=1}^{n_f} Adapter_{F2}(f_{i,j}), \quad (10)$$

so that we can process this video embedding sequence $[v_i^0]_{i=1}^{n_v}$.

Video-Identity-Text Alignment

In order to achieve rapid video retrieval, it is necessary to process the representations of both the video and query separately and align them. Specifically, Video-Locator employs the first L_u layers of the LLM to integrate contextual information from both video and text respectively. For the video embedding sequence, the process is as follows:

$$[v_i^l]_{i=1}^{n_v} = Decoder_l([v_i^{l-1}]_{i=1}^{n_v}), \ \ l = 1, 2, ..., L_u. \quad (11)$$

As for the query Q, we append a learnable embedding x_{cls} at the end of the query embedding sequence to compute the sentence-level representation. This sequence is also processed through the first L_u layers of the LLM as follows:

$$[x_i^l]_{i=1}^{n_q+1} = Decoder_l([x_i^{l-1}]_{i=1}^{n_q+1}), \ \ l = 1, 2, ..., L_u, \quad (12)$$

where $x_{n_q+1}^0 = x_{cls}$, and $x_{cls}^{L_u} = x_{n_q+1}^{L_u}$ serves as the representation of the query. At this point, we can define the similarity between the query Q and video V, denoted as $sim(Q, V)$, as the maximum cosine similarity between the text and each frame of the video:

$$sim(Q, V) = \max_{i=1}^{n_v}\{\cos(x_{cls}^{L_u}, v_i^{L_u})\}. \quad (13)$$

We employ the contrastive loss (InfoNCE(Oord, Li, and Vinyals 2018)) to increase the similarity between matching video-text pairs and decrease the similarity between non-matching pairs. Specifically, given a batch of queries and corresponding texts $\{(Q_1, V_1), (Q_2, V_2), ..., (Q_B, V_B)\}$, the contrastive loss $\mathcal{L}_{\text{cl}}$ is defined as follows:

$$\mathcal{L}_{\text{q2v}} = -\sum_{i=1}^{B}\log\frac{\exp(sim(Q_i, V_i)/\tau)}{\sum_{j=1}^{B}\exp(sim(Q_i, V_j)/\tau)}, \quad (14)$$

$$\mathcal{L}_{\text{v2q}} = -\sum_{i=1}^{B}\log\frac{\exp(sim(Q_i, V_i)/\tau)}{\sum_{j=1}^{B}\exp(sim(Q_j, V_i)/\tau)}, \quad (15)$$

$$\mathcal{L}_{\text{cl}} = (\mathcal{L}_{\text{q2v}} + \mathcal{L}_{\text{v2q}})/2, \quad (16)$$

where τ is a temperature parameter, B is the batch size.

Multi-Modal Fine-Grained Fusion

After the video-identity-text alignment, we can find the matched video for the given text and identity images. Next, we will locate the fine-grained moments in the video for the given query. Specifically, for the matched query and video, we will fuse their embeddings and provide the final predicted moments. Video-Locator leverages the top $L - L_u$ layers of the LLM as a multi-modal fine-grained fusion module, where we concatenate the query embedding sequence with the video embedding sequence and append a learnable embedding x_{ground} at the end. This sequence is then processed through the top $L - L_u$ layers as follows:

$$\begin{aligned}&[x_1^l, ..., x_{n_q+1}^l, v_1^l, ..., v_{n_v}^l, x_{ground}^l] \\ &= Decoder_l([x_1^{l-1}, ..., x_{n_q+1}^{l-1}, v_1^{l-1}, ..., v_{n_v}^{l-1}, x_{ground}^{l-1}]), \\ &l = L_u + 1, L_u + 2, ..., L,\end{aligned} \quad (17)$$

where $x_{ground}^{L_u} = x_{ground}$. Finally, we employ a 3-layer MLP, denoted as $Ground(\cdot)$, to output the predicted moment $(\hat{\tau}^s, \hat{\tau}^t)$ as follows:

$$(\hat{\tau}^s, \hat{\tau}^t) = Ground(x_{ground}^L). \quad (18)$$

We employ L1 loss and generalized IoU loss to compute the disparity between the predicted moment and the ground truth moment (τ^s, τ^t):

$$\mathcal{L}_{l1} = ||\tau^s - \hat{\tau}^s|| + ||\tau^t - \hat{\tau}^t||, \quad (19)$$

Table 2: Results of Video-Locator and other models on the TVR-IT dataset.

Model	test-seen							test-unseen						
	VCG		VR			SVG		VCG		VR			SVG	
	0.5	0.7	R@1	R@10	R@100	0.5	0.7	0.5	0.7	R@1	R@10	R@100	0.5	0.7
XML	7.55	4.13	28.08	64.82	94.06	27.67	13.89	1.70	0.83	6.82	26.18	62.41	23.71	11.05
ReLoCLNet	8.29	3.95	29.34	66.06	94.08	26.43	12.18	2.23	1.15	7.78	27.75	62.07	23.63	11.01
SQuiDNet	12.30	5.97	34.93	65.42	94.51	35.45	18.19	3.27	1.81	9.12	28.73	62.75	29.43	15.32
Video-Locator	**16.86**	**8.32**	**36.04**	**73.55**	**96.23**	**41.19**	**20.07**	**4.64**	**2.27**	**9.97**	**31.83**	**64.74**	**36.95**	**17.43**

while the generalized IoU loss $\mathcal{L}_{giou}$ is defined in (Rezatofighi et al. 2019).

Finally, we introduce an auxiliary loss to enhance the model's performance. Specifically, we employ a linear layer $Classifier(\cdot)$ to compute the probability of each frame falling within the ground truth moment and optimize this using binary cross entropy loss as follows:

$$\hat{y}_i = Classifier(v_i^L), \tag{20}$$

$$\mathcal{L}_{bce} = -\frac{1}{n_v}\sum_{i=1}^{n_v}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)], \tag{21}$$

where y_i represents a binary variable denoting whether the i-th frame falls within the ground truth moment. The overall loss is defined as follows:

$$\mathcal{L} = \lambda_{cl}\mathcal{L}_{cl} + \lambda_{l1}\mathcal{L}_{l1} + \lambda_{giou}\mathcal{L}_{giou} + \lambda_{bce}\mathcal{L}_{bce}. \tag{22}$$

Inference

During inference, given a query Q and a video corpus $\mathcal{V}$, we independently compute the representations for the query and each video. Subsequently, we employ Eq.(13) to calculate the similarity between the query and each video, selecting the video-text pair with the highest similarity as the matched one. The matched video-text pair is then processed through the multi-modal fine-grained fusion module to output the predicted moments. When dealing with N queries, this approach requires only $N+|\mathcal{V}|$ representation computations and N multi-modal fine-grained fusion computations, thereby avoiding the need to compute fusion representations for all possible pairs, resulting in a substantial efficiency enhancement.

Experiment

Experiment Setup

Implementation Details For the Video-Locator model, we employ Vicuna-v1.5-7B (Chiang et al. 2023) as the LLM, which comprises $L = 32$ layers. $L_u = 16$ layers are dedicated to the video-identity-text alignment module, while the remaining 16 layers are utilized for the multi-modal fine-grained fusion module. For each video, we uniformly sample $n_v = 100$ frames, retaining a maximum of $n_f = 3$ faces per frame. The temperature parameter τ is set to 0.07. We balance the losses using parameters $\lambda_{cl} = 1, \lambda_{l1} = 10, \lambda_{giou} = 1, \lambda_{bce} = 4$, respectively. We utilize LoRA (Hu et al. 2022) for training the LLM decoders of our Video-Locator to minimize the consumption of training resources. The LoRA rank parameters are set to $r = 64$.

Tasks and Evaluation Metrics As video corpus grounding can be considered a multi-stage task, we follow previous works(Escorcia et al. 2019; Lei et al. 2020) and evaluate our approach on video retrieval (VR), single video grounding (SVG), and complete video corpus grounding (VCG) tasks. Specifically, for VR, we employ the same metrics as those used for text-to-video retrieval, namely $R@n$ (Recall at n, $n = 1, 10, 100$), which measures the proportion of correct videos within the top n retrieved videos. We use the same metrics for VCG and SVG, calculating the Intersection over Union (IoU) between the generated and ground truth time segments. We report the IoU m metric, with $m = 0.5, 0.7$, representing the proportion of $IoU \geq m$. The difference between VCG and SVG is that in VCG, we use the retrieved videos from the model to calculate IoU m (if the retrieved video is wrong then the IoU is 0), but in SVG, we directly use the ground-truth video to conduct video grounding to calculate its IoU.

Main Results

We selected several open-source VCG models, including XML(Lei et al. 2020), ReLoCLNet(Zhang et al. 2021), and SQuidNet(Yoon et al. 2022), and trained on the TVR-IT dataset using their official GitHub repositories. We used the same settings as Video-Locator: visual features (CLIP + Antelopev2), without subtitles, and modified the input part of the model (added facial adapters) to accept face inputs similar to Video-Locator. We reported the metrics in Table 2. Our Video-Locator significantly outperforms the other models. It is also noteworthy that other models require location of the moment within the top-10 retrieved videos and subsequent reordering of all moments. In contrast, our approach requires location only within the top-1 video, resulting in significant efficiency gains.

Ablation Study

In this section, we provide detailed ablations on the Video-Locator model, as illustrated in Table 3. In Video-Locator, we utilize an image-text dataset to pretrain the visual adapter by generating text from given images (Row 1). Compared to models without pretraining (Row 2), the pretrained model exhibits superior performance across all metrics. The results from Row 3 to 6 in the table demonstrate that the Video-Locator model with all four losses performs best in the VCG task, regardless of whether it is evaluated on the seen or unseen split. Specifically, removing $\mathcal{L}_{cl}$ results in a lack of alignment between the video and query, leading to nearly zero performance in the VR task. The removal of any of the

Table 3: Ablation study of the Video-Locator model.

Row	Model	test-seen							test-unseen						
		VCG		VR			SVG		VCG		VR			SVG	
		0.5	0.7	R@1	R@10	R@100	0.5	0.7	0.5	0.7	R@1	R@10	R@100	0.5	0.7
1	Video-Locator	**16.86**	**8.32**	**36.04**	**73.55**	**96.23**	**41.19**	20.07	**4.64**	**2.27**	**9.97**	**31.83**	**64.74**	**36.95**	**17.43**
2	w/o pretrain	14.29	7.22	31.33	70.44	95.39	40.12	19.95	3.58	1.69	6.95	25.16	61.14	34.88	15.37
3	w/o $\mathcal{L}_{cl}$	0.02	0.02	0.04	0.78	7.51	40.74	**20.83**	0.00	0.00	0.11	0.83	7.65	36.60	17.32
4	w/o $\mathcal{L}_{l1}$	15.32	7.78	32.15	70.87	94.90	38.43	19.68	3.68	1.78	7.81	26.83	60.19	29.33	13.38
5	w/o $\mathcal{L}_{giou}$	13.48	6.03	32.98	70.73	95.35	33.22	15.73	3.56	1.65	8.85	28.10	63.41	26.56	10.78
6	w/o $\mathcal{L}_{bce}$	16.78	8.25	35.63	72.77	95.78	40.28	18.96	4.19	2.03	8.66	28.95	63.28	36.65	17.41

Table 4: Results of Video-Locator-T and the Re-rank method.

Model	test-seen							test-unseen						
	VCG		VR			SVG		VCG		VR			SVG	
	0.5	0.7	R@1	R@10	R@100	0.5	0.7	0.5	0.7	R@1	R@10	R@100	0.5	0.7
Video-Locator	16.86	8.32	36.04	**73.55**	**96.23**	**41.19**	**20.07**	4.64	2.27	9.97	31.83	64.74	36.95	17.43
Video-Locator + Re-rank	**17.35**	**8.56**	**37.12**	72.20	**96.23**	**41.19**	**20.07**	6.27	3.28	12.79	36.98	64.74	36.95	17.43
Video-Locator-T	7.12	3.35	16.45	46.11	79.66	38.27	18.81	6.29	3.30	13.13	39.26	**72.91**	**39.64**	**19.17**
Video-Locator-T + Re-rank	12.70	6.11	29.18	61.24	79.66	38.27	18.81	**8.15**	**4.07**	**17.47**	**45.53**	**72.91**	**39.64**	**19.17**

remaining three losses also negatively impacts all tasks, even though they were designed for video grounding. This is because video grounding involves aligning video and query at a finer granularity (frame level), which also benefits video retrieval.

Future Work: Identity Generalization

We found that all models including Video-Locator exhibit poor generalization regarding identities, as the performance on the unseen split is significantly worse than on the seen split. This is due to the fact that only hundreds of identities are collected from only six TV shows, causing the model to overfit to these identities. However, addressing this issue proves challenging. For instance, we attempted to use the large-scale face dataset CelebA(Liu et al. 2018) for pre-training, but faces are not suitable for LLM pre-training, because face features are challenging to describe in language. We also considered pre-training with identity-related videos. However, finding a suitable dataset is difficult. This remains a promising direction for future.

We propose a solution called Re-rank, which does not modify the model's training process and only operates during inference. Specifically, after the model retrieves the top 100 potential matched videos, we re-rank them based on the number of identities from the query that appear in these videos. (Previously we only use the embedding similarity between videos and queries to rank the videos.) A video with more identities from the query is ranked higher. If two videos contain the same number of identities, the original ranking is used. To determine whether a identity appears in a video, we compute the similarity between their face embedding and the face embeddings of all faces appearing in the video (Eq. 1). If there exists a similarity greater than the threshold of $\theta = 0.4$, we consider the identity to be present. The Re-rank method affects only the R@1 and R@10 metrics in the VR task and the VCG task.

Additionally, we train a model called Video-Locator-T that does not incorporate face input. On the video side, it does not introduce facial feature, and on the query side, it replaces the identity image with the word *'someone'*, resulting in a purely textual query. Table 4 shows the results for Video-Locator and Video-Locator-T, as well as their results after applying the Re-rank method. We find that:

- Video-Locator outperforms Video-Locator-T on the test-seen split, but performs worse than Video-Locator-T on the test-unseen split. This suggests that Video-Locator can only recognize identities from the seen dataset and lacks the ability to generalize to unseen identities.
- Re-rank method significantly improves the model's performance on unseen identities, indicating that identity images can provide a substantial amount of information.

We look forward to future research addressing the issue of identity generalization. This may involve expanding the training dataset to expose the model to a greater variety of identities, or implementing model designs or training methods specifically aimed at enhancing generalization ability.

Conclusion

This paper introduces a novel and significant task called Identity-Text Video Corpus Grounding (ITVCG), which goes beyond traditional VCG by using specific identity image references alongside text descriptions to find relevant video content. To support this task, we have created the TVR-IT dataset, containing 463 identities and 72,840 identity-text queries, with both identity-seen and identity-unseen splits to test model generalization. We propose the Video-Locator model, which includes a Video-Identity-Text Alignment module and a Multi-Modal Fine-Grained Fusion module. Our experiments show that the Video-Locator performs well and highlight the importance of identity generalization in ITVCG.

Acknowledgments

This work was supported by the National Key Research and Development Program of China No.2023YFF1205001, National Natural Science Foundation of China No. 62222209, Beijing National Research Center for Information Science and Technology under Grant No. BNR2023TD03006, BNR2023RC01003, and Beijing Key Lab of Networked Multimedia.

References

Anne Hendricks, L.; Wang, O.; Shechtman, E.; Sivic, J.; Darrell, T.; and Russell, B. 2017. Localizing moments in video with natural language. In *Proceedings of the IEEE international conference on computer vision*, 5803–5812.

Chen, H.; Wang, X.; Chen, H.; Song, Z.; Jia, J.; and Zhu, W. 2023a. Grounding-Prompter: Prompting LLM with Multimodal Information for Temporal Sentence Grounding in Long Videos. *arXiv preprint arXiv:2312.17117*.

Chen, H.; Wang, X.; Chen, H.; Zhang, Z.; Feng, W.; Huang, B.; Jia, J.; and Zhu, W. 2024a. VERIFIED: A Video Corpus Moment Retrieval Benchmark for Fine-Grained Video Understanding. *arXiv preprint arXiv:2410.08593*.

Chen, H.; Wang, X.; Lan, X.; Chen, H.; Duan, X.; Jia, J.; and Zhu, W. 2023b. Curriculum-listener: Consistency-and complementarity-aware audio-enhanced temporal sentence grounding. In *Proceedings of the 31st ACM International Conference on Multimedia*, 3117–3128.

Chen, H.; Wang, X.; Zhou, Y.; Huang, B.; Zhang, Y.; Feng, W.; Chen, H.; Zhang, Z.; Tang, S.; and Zhu, W. 2024b. Multi-modal generative ai: Multi-modal llm, diffusion and beyond. *arXiv preprint arXiv:2409.14993*.

Chiang, W.-L.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J. E.; Stoica, I.; and Xing, E. P. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Dong, J.; Chen, X.; Zhang, M.; Yang, X.; Chen, S.; Li, X.; and Wang, X. 2022. Partially relevant video retrieval. In *Proceedings of the 30th ACM International Conference on Multimedia*, 246–257.

Escorcia, V.; Soldan, M.; Sivic, J.; Ghanem, B.; and Russell, B. 2019. Temporal localization of moments in video collections with natural language.

Feng, W.; Wang, X.; Chen, H.; Zhang, Z.; Chen, H.; Song, Z.; Zhou, Y.; Yang, Y.; Wu, H.; and Zhu, W. 2023. Llm4vg: Large language models evaluation for video grounding. *arXiv preprint arXiv:2312.14206*.

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

Hou, D.; Pang, L.; Shen, H.; and Cheng, X. 2024. Event-aware Video Corpus Moment Retrieval. *arXiv preprint arXiv:2402.13566*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Huang, B.; Wang, X.; Chen, H.; Song, Z.; and Zhu, W. 2024. Vtimellm: Empower llm to grasp video moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14271–14280.

Lei, J.; Yu, L.; Berg, T. L.; and Bansal, M. 2020. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, 447–463. Springer.

Li, L.; Chen, Y.-C.; Cheng, Y.; Gan, Z.; Yu, L.; and Liu, J. 2020. HERO: Hierarchical Encoder for Video+ Language Omni-representation Pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2046–2065.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2018. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018): 11.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 658–666.

Wang, X.; Lan, X.; and Zhu, W. 2022. Video grounding and its generalization. In *Proceedings of the 30th ACM International Conference on Multimedia*, 7377–7379.

Wang, X.; Wu, Z.; Chen, H.; Lan, X.; and Zhu, W. 2023. Mixup-augmented temporally debiased video grounding with content-location disentanglement. In *Proceedings of the 31st ACM International Conference on Multimedia*, 4450–4459.

Yoon, S.; Hong, J. W.; Yoon, E.; Kim, D.; Kim, J.; Yoon, H. S.; and Yoo, C. D. 2022. Selective query-guided debiasing for video corpus moment retrieval. In *European Conference on Computer Vision*, 185–200. Springer.

Yu, Z.; Xu, D.; Yu, J.; Yu, T.; Zhao, Z.; Zhuang, Y.; and Tao, D. 2019. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 9127–9134.

Yuan, Y.; Lan, X.; Wang, X.; Chen, L.; Wang, Z.; and Zhu, W. 2021. A closer look at temporal sentence grounding in videos: Dataset and metric. In *Proceedings of the 2nd international workshop on human-centric multimedia analysis*, 13–21.

Zhang, H.; Sun, A.; Jing, W.; Nan, G.; Zhen, L.; Zhou, J. T.; and Goh, R. S. M. 2021. Video corpus moment retrieval

with contrastive learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 685–695.