

# Moment Quantization for Video Temporal Grounding

Xiaolong Sun<sup>1</sup> Le Wang<sup>1\*</sup> Sanping Zhou<sup>1</sup> Liushuai Shi<sup>1</sup>

Kun Xia<sup>2</sup> Mengnan Liu<sup>1</sup> Yabing Wang<sup>1</sup> Gang Hua<sup>3</sup>

<sup>1</sup>National Key Laboratory of Human-Machine Hybrid Augmented Intelligence,  
National Engineering Research Center for Visual Information and Applications,

Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University

<sup>2</sup>School of Computer Science and Technology, Xi'an Jiaotong University

<sup>3</sup>Amazon Alexa AI

## Abstract

*Video temporal grounding is a critical video understanding task, which aims to localize moments relevant to a language description. The challenge of this task lies in distinguishing relevant and irrelevant moments. Previous methods focused on learning continuous features exhibit weak differentiation between foreground and background features. In this paper, we propose a novel **Moment-Quantization based Video Temporal Grounding method (MQVTG)**, which quantizes the input video into various discrete vectors to enhance the discrimination between relevant and irrelevant moments. Specifically, MQVTG maintains a learnable moment codebook, where each video moment matches a codeword. Considering the visual diversity, i.e., various visual expressions for the same moment, MQVTG treats moment-codeword matching as a clustering process without using discrete vectors, avoiding the loss of useful information from direct hard quantization. Additionally, we employ effective prior-initialization and joint-projection strategies to enhance the maintained moment codebook. With its simple implementation, the proposed method can be integrated into existing temporal grounding models as a plug-and-play component. Extensive experiments on six popular benchmarks demonstrate the effectiveness and generalizability of MQVTG, significantly outperforming state-of-the-art methods. Further qualitative analysis shows that our method effectively groups relevant features and separates irrelevant ones, aligning with our goal of enhancing discrimination. Code is available at <https://github.com/TensorsSun/MQVTG>.*

## 1. Introduction

The rapid expansion of short-form video content on social media platforms has made video a highly engaging multi-

medium format [1, 19]. The significant surge prompts users to selectively engage with brief moments of interest rather than passively watch entire videos. This derives the topic of video temporal grounding (VTG), which aims to ground video clips based on natural language descriptions. Aside from the moment retrieval [6, 13, 16], recent works in VTG have also been interested in highlight detection [15, 35]. As a multi-modal video understanding task, variable and open-vocabulary language descriptions complicate video representation learning in VTG. This makes it difficult to distinguish between relevant and irrelevant moments, posing significant challenges to the task.

To address this challenge, previous works [11, 34, 36] learn continuous features with complex modal interactions. However, videos contain much more redundant information, while VTG aims to separate video clips into foreground and background, focusing on essential discriminative information. As shown in Fig. 1(a), we present an example with multiple non-contiguous foreground moments. Due to video information redundancy, previous methods (e.g., TR-DETR [27]) struggle with distinguishing foreground and similar background features (i.e., red and blue points are close in feature space) and poorly grouping foreground features (i.e., different red points are far apart in feature space). Additionally, the interested foreground can be described concisely by a discrete language query [32, 33]. For example in Fig. 1(b), we can accurately describe the video moment of a spoon stirring curry using the discrete language query. That motivates us to ask: *Can we describe the continuous video moments via discrete vectors to enhance discrimination between relevant and irrelevant moments?* Existing works [5, 38, 39] have explored various patch-level quantization on images, however, moment-level quantization on videos remains an unexplored area.

To achieve our goal, we propose a **Moment-Quantization based Video Temporal Grounding method (MQVTG)**, which quantizes the video moments into discrete vectors.

\*Corresponding author.

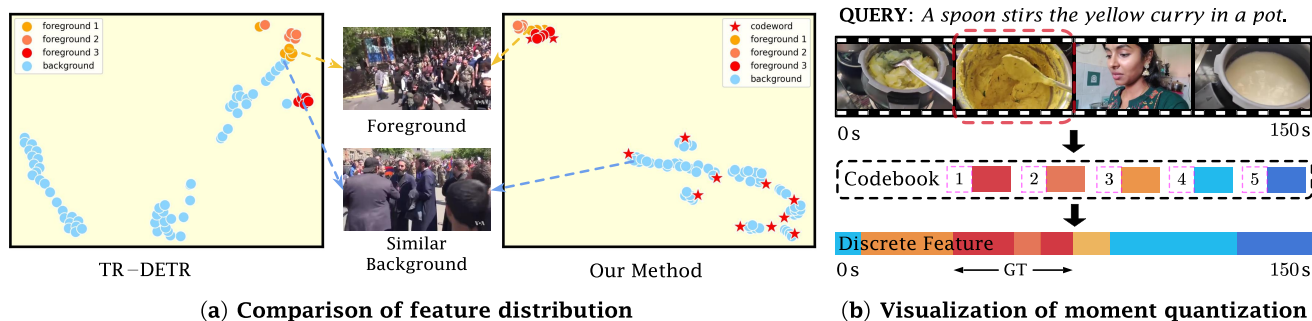


Figure 1. (a) Comparison of codebook vectors, foreground and background features between TR-DETR [27] and our method for a given example. Compared to previous methods focused on learning continuous features, our method, aided by codebook vectors, achieves better foreground aggregation and foreground-background separation. (b) Visualization of moment quantization. The moment quantization discriminates foreground and background moments. The foregrounds are represented by red-related vectors, while the backgrounds are represented by other discrete vectors. The quantized features bring discriminative information to generate more accurate localization.

The foreground and background are distinguished by different vectors, thus improving the discrimination between relevant and irrelevant moments. As shown in Fig. 1(b), the foreground moments are represented by red-related vectors, while the background moments are represented by other discrete vectors. To construct this moment quantization, a simple implementation, named clip quantization, is proposed to quantize individual video clips into a discrete codebook similar to traditional patch quantization on images. However, clip quantization overlooks the characteristics of the video moments: 1) A moment crosses multiple video clips (cross-clip); 2) A moment described by the same language may express various visual forms (visual diversity).

Drive by the two characteristics, the moment quantization is built on the clip quantization with three innovations. First, we maintain the moment codebook based on the video clips after temporal modeling rather than isolated video clips to meet the cross-clip nature of the moment. Second, due to the visual diversity of moments, directly using discrete codewords loses distinctive information. For example in Fig. 1(a), directly replacing all video features with limited codebook vectors is clearly inappropriate. Thus, we deliver the quantized continuous features into the downstream localization module, to retain the visual diversity. Furthermore, two effective prior initialization and joint projection strategies are proposed to enhance the moment codebook.

We conduct extensive experiments on six popular video temporal grounding benchmarks to validate the effectiveness of our method, which achieves state-of-the-art performances for all benchmarks. The moment quantization also serves as a plug-and-play component, performing well in both encoder-only and encoder-decoder architectures. As shown in Fig. 1(a), our method can effectively group foreground and separate fore/background features, aligning with our goal of enhancing discrimination. Our main contributions are summarized as follows:

- To our knowledge, this is the first introduction of vector quantization to the VTG task. We propose a Moment-Quantization based Video Temporal Grounding method that describes the continuous video moments via discrete vectors to enhance discrimination between moments.
- To adapt vector quantization from images to videos, we introduce two progressive implementations, clip quantization and moment quantization, both capable of quantizing video features to capture discriminative information.
- Clip quantization, as a naive implementation, simply aligns with image quantization, while moment quantization considers the characteristics of video moments, *i.e.*, cross-clip and visual diversity.
- Extensive experiments demonstrate the effectiveness of our method, and it can be integrated into existing grounding models, showing strong generalizability.

## 2. Related Work

**Video Temporal Grounding.** Video temporal grounding can be divided into moment retrieval (MR) [6, 10, 13, 21] and highlight detection (HD) [2, 9, 28], which localizes the relevant moments and scores the clip-wise correspondence to the query. Recent research [13] constructs the QVHighlights dataset to facilitate joint learning of MR&HD and proposes a baseline model based on DETR. QD-DETR [21] exploits the textual information by involving video-text pair negative relationship learning. R<sup>2</sup>-Tuning [17] learns a lightweight side adapter to adaptively pool spatial details and refine temporal correlations. However, all previous methods focus on learning continuous features, overlooking the inherently discrete semantic nature of video moments. In contrast, we introduce the discrete learning approach of vector quantization to aid the VTG task.

**Vector Quantization.** Vector quantization aims to represent the data with entries of a learnable codebook, which

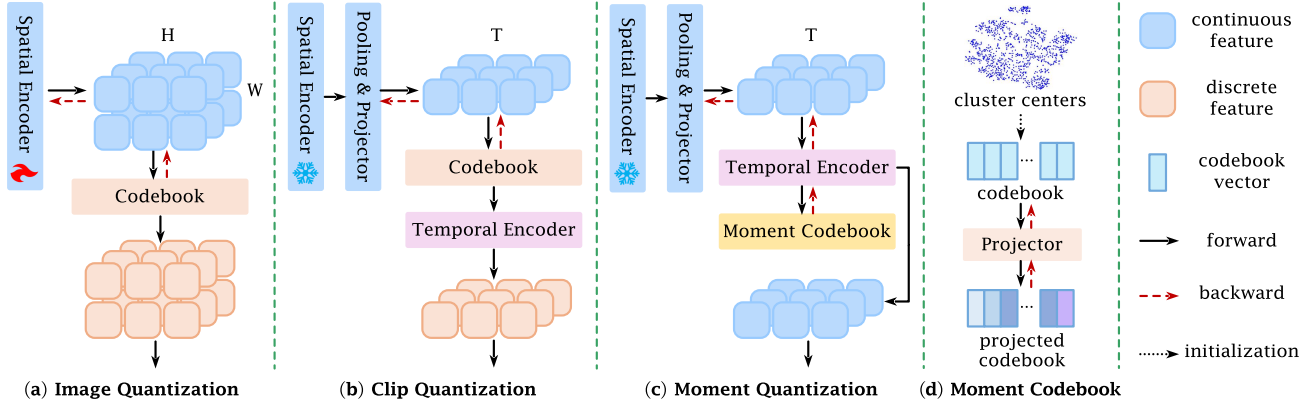


Figure 2. Comparison of three quantization methods including (a) the classic image quantization, (b) clip quantization that is a simple implementation of vector quantization for videos, and (c) the improved moment quantization for video temporal grounding. The design of the moment codebook used in moment quantization is shown in (d).

achieves discrete and compressed representation. Many researchers [4, 18, 22] have shown that learning discrete representation contributes to visual understanding. Recently, VQVAE [23] uses a codebook to learn a discrete visual representation of images, which can learn the discrete feature distribution of an image effectively and is widely used in many generative models. Along this line of research, discrete representation learning has been widely used in many vision tasks [5, 8, 37, 39]. Existing works have explored various quantization methods on images or audio, however, vector quantization for videos remains an unexplored area.

### 3. Method

Given an untrimmed video  $\mathcal{V}$  with  $T$  frames and an associated natural language description  $\mathcal{Q}$  with  $N$  words, video temporal grounding aims to predict clip-wise saliency scores and localize the center coordinate and span of target moments that are most relevant to the description. Our goal is to introduce the discrete learning approach in vector quantization to video temporal grounding, aiming to enhance discrimination between relevant and irrelevant moments. Thus we propose a moment quantization method for video temporal grounding, called MQVTG.

In this section, we first introduce the preliminaries of image quantization in Sec. 3.1. Then, we propose the clip quantization in Sec. 3.2, which is a simple implementation of vector quantization for videos. The improved moment quantization and its moment codebook are explained in detail in Sec. 3.3 and Sec. 3.4. Finally, the overall architecture and training objectives of our method are introduced in Sec. 3.5 and Sec. 3.6, respectively.

#### 3.1. Preliminaries: Image Quantization

An image quantization model is typically a reconstructive encoder-decoder architecture that includes a codebook

module to convert continuous representations into discrete ones as illustrated in Fig. 2 (a). Formally, the spatial encoder maps the input image into a latent space, producing a continuous feature  $z \in \mathbb{R}^{H \times W \times d}$ , where  $d$  denotes the hidden dimension. The feature is then quantized using a learnable codebook  $C \in \mathbb{R}^{K \times d}$ , where  $K$  represents the number of codewords in the codebook. The codebook module selects the nearest codeword by minimizing the distance between  $z$  and the codewords:

$$\hat{z} = C(z) = c_k, \text{ where } k = \arg \min_i \|z - c_i\|_2^2, \quad (1)$$

where  $c_i$  is the  $i$ -th codeword.  $c_k$  is usually denoted by  $\hat{z}$ , that is, the quantized discrete feature passed to the decoder to reconstruct the input image. The most classic image quantization model is VQ-VAE [31], which uses a straight-through estimator [3] with codebook loss and commitment loss to narrow the gap between the selected codewords and the encoder output. Image quantization has been widely studied [5, 38, 39], however, vector quantization on videos remains an unexplored area.

#### 3.2. Clip Quantization

From the above perspective, the codebook update of the vector quantization model is intrinsically a dictionary learning process. In other words, the codebook training is like  $k$ -means clustering, where cluster centers are the discrete codewords. The feature discrimination formed by the clustering process aligns with the requirement of the VTG task to distinguish between relevant and irrelevant moments. Based on this, we propose two progressive implementations of vector quantization for videos: clip and moment quantization. Both of them quantize video features to capture discriminative information, while clip quantization simply aligns with image quantization and moment quantization

considers the cross-clip nature and visual diversity of video moments. We first introduce clip quantization in Fig. 2 (b).

**Quantization before Temporal Modeling.** Unlike previous image quantization, which focuses on spatial patches, we aim to perform quantization on temporal video clips. As shown in Fig. 2 (b), after passing through a frozen spatial encoder, a pooling layer and a linear projector, we obtain the pooled visual feature  $z_s \in \mathbb{R}^{T \times d}$ , where  $T$  is the number of video clips. Subsequently, a codebook module is used to quantize the continuous visual feature  $z_s$  by Eq. 1, which is supervised by the codebook loss  $\mathcal{L}_{cb}$  and the commitment loss  $\mathcal{L}_{cmt}$  [31]. The quantized discrete feature is then sent to a multi-modal temporal encoder  $E_t$  to generate the video feature for temporal grounding. For simplicity, we omit the text input and the multi-modal interaction process.

The difference between clip and image quantization is that the former quantizes individual video clips, while the latter quantizes spatial image patches. By optimizing the projector after the frozen spatial encoder with the codebook and commitment loss, clip quantization enables feature clustering for these individual video clips. However, as a simple implementation of vector quantization for videos, it lacks careful consideration for video moments. Therefore, we propose a more comprehensive quantization method for videos in the next section, called moment quantization.

### 3.3. Moment Quantization

The implementation of our moment quantization is illustrated in Fig. 2 (c). Compared to the simple clip quantization, it introduces three main improvements: 1) We choose to apply quantization after temporal modeling to meet the cross-clip nature of moments; 2) We introduce a soft quantization operation instead of directly using discrete features to accommodate visual diversity; 3) We propose a moment codebook to better adapt moment quantization. In this section, we first explain the first two points, and the moment codebook module will be discussed in Sec. 3.4.

**Quantization after Temporal Modeling.** The clip quantization quantizes video clips before the temporal encoder  $E_t$ . Without temporal modeling, these video clips remain isolated and fail to form a complete moment—representing a full action or event, which conflicts with the cross-clip nature of moments. Therefore, moment quantization is applied after temporal modeling. As shown in Fig. 2 (c), the pooled visual feature  $z_s$  is first sent to the temporal encoder  $E_t$  to generate the semantic-aware continuous video feature  $z_t = E_t(z_s)$ . Subsequently, we use the moment codebook module (discussed in Sec. 3.4) to quantize the continuous feature  $z_t$ . The process is supervised by the codebook loss  $\mathcal{L}_{cb}$  and the commitment loss  $\mathcal{L}_{cmt}$  [31].  $\mathcal{L}_{cb}$  for updating the codebook parameters is as follows:

$$\mathcal{L}_{cb} = \|\hat{z}_t - \text{sg}(z_t)\|_2^2 = \|C(z_t) - \text{sg}(E_t(z_s))\|_2^2, \quad (2)$$

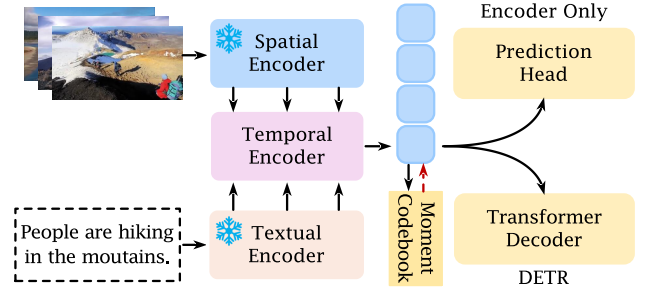


Figure 3. The architectures of MQVTG, including the encoder-only architecture and encoder-decoder (DETR) architecture.

where  $\text{sg}(\cdot)$  represents the stop-gradient operation and  $\hat{z}_t$  is the quantized discrete vector from the codebook. Similarly,  $\mathcal{L}_{cmt}$  can be formulated as:

$$\mathcal{L}_{cmt} = \|\text{sg}(\hat{z}_t) - z_t\|_2^2 = \|\text{sg}(C(z_t)) - E_t(z_s)\|_2^2, \quad (3)$$

which is used to optimize the parameters in the temporal encoder  $E_t$  and make the output of  $E_t$  consistent with the codebook embedding space. The two losses jointly guide the feature-codeword clustering process.

**Soft Quantization with Continuous Features.** Image quantization usually replaces the continuous feature with the quantized discrete vector directly and inputs it into the subsequent decoder. However, videos have greater visual diversity and information density compared to images, which means that the visual presentation of the same language description can vary greatly in video format. In this case, the discrete vectors from a limited-capacity codebook cannot accurately represent an entire video as they do for an image. On the other hand, due to the direct replacement operation in Eq. 1 and the poorly optimized codebook in the early training stage, some useful information may inevitably be lost [37], thus affecting the final localization. We will explain this in detail in Sec. 4.6. Based on these considerations, we introduce a soft quantization operation. As shown in Fig. 2 (c), we continue using the continuous features  $z_t$  instead of the quantized discrete vector  $\hat{z}_t$  for subsequent localization. We argue that the feature-codeword clustering process, driven by Eq. 2 and Eq. 3, enables the continuous feature  $z_t$  to learn discriminative information, whereas directly using discrete vectors may actually be detrimental. The designed ablation experiments in Sec. 4.5 support our perspective.

### 3.4. Moment Codebook

In the previous image quantization models [5, 31, 38], the codebook vectors are initialized randomly. As shown in Eq. 1, during each iteration, only a small subset of codebook vectors related to the current training batch are optimized. Therefore, the random initialization causes only

Method	QVHighlights-val						QVHighlights-test							
	MR-R1		MR-mAP			HD		MR-R1		MR-mAP			HD	
	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1	@0.5	@0.7	@0.5	@0.75	Avg.	mAP	HIT@1
M-DETR [13] <i>NeurIPS'21</i>	53.94	34.84	—	—	32.20	35.65	55.55	52.89	33.02	54.82	29.40	30.73	35.69	55.60
QD-DETR [21] <i>CVPR'23</i>	62.68	46.66	62.23	41.82	41.22	39.13	63.03	62.40	44.98	62.52	39.88	39.86	38.94	62.40
UniVTG [15] <i>ICCV'23</i>	59.74	—	—	—	36.13	38.83	61.81	58.86	40.86	57.60	35.59	35.47	38.20	60.96
TR-DETR [27] <i>AAAI'24</i>	67.10	51.48	66.27	46.42	45.09	40.55	64.77	64.66	48.96	63.98	43.73	42.62	39.91	63.42
CG-DETR [20] <i>Arxiv'24</i>	67.35	52.06	65.57	45.73	44.93	<b>40.79</b>	<b>66.71</b>	65.43	48.38	64.51	42.77	42.86	40.33	<b>66.21</b>
UVCOM [34] <i>CVPR'24</i>	65.10	51.81	—	—	45.79	40.03	63.29	63.55	47.47	63.37	42.67	43.18	39.74	64.20
TaskWeave [36] <i>CVPR'24</i>	64.26	50.06	65.39	46.47	45.38	39.28	63.68	—	—	—	—	—	—	—
R <sup>2</sup> -Tuning [17] <i>ECCV'24</i>	<b>68.71</b>	52.06	—	—	47.59	<u>40.59</u>	64.32	<b>68.03</b>	49.35	<b>69.04</b>	<u>47.56</u>	<u>46.17</u>	<b>40.75</b>	64.20
BAM-DETR [12] <i>ECCV'24</i>	65.10	51.61	65.41	<u>48.56</u>	<u>47.61</u>	—	—	62.71	48.64	64.57	46.33	45.36	—	—
LLMEPET [11] <i>ACMMM'24</i>	66.58	51.10	—	—	46.24	—	—	<u>66.73</u>	<u>49.94</u>	65.76	43.91	44.05	<u>40.33</u>	<u>65.69</u>
SpikeMba [14] <i>Arxiv'24</i>	65.32	51.33	—	44.96	44.84	—	—	64.13	49.42	—	43.67	43.79	—	—
RGTR [29] <i>AAAI'25</i>	67.68	<u>52.90</u>	<u>67.38</u>	48.00	46.95	—	—	65.50	49.22	67.12	45.77	45.53	—	—
<b>MQVTG (Ours)</b>	<u>67.94</u>	<b>53.03</b>	<b>68.54</b>	<b>51.48</b>	<b>48.81</b>	40.23	<u>65.29</u>	66.28	<b>50.00</b>	<u>67.98</u>	<b>48.55</b>	<b>47.08</b>	39.49	64.07

Table 1. Moment retrieval (MR) and highlight detection (HD) results on QVHighlights *val* and *test* splits.

a few frequently optimized codebook vectors to align with the feature distribution generated by the temporal encoder. This conflicts with the rich semantic representation needed for video moments. On the other hand, different codebook vectors are independent of each other in previous methods. However, the video clips from the same moment are semantically related. We establish this temporal semantic correlation via the temporal encoder and believe it should extend to the codebook vectors as well. Based on the above discussion, we propose a moment codebook, as illustrated in Fig. 2 (d), which employs effective prior-initialization and joint-projection strategies to adapt moment quantization.

**Prior Initialization.** We first extract patch-level features for each clip in each video from the training dataset via a pre-trained visual encoder (*e.g.*, the CLIP model), and then obtain the clip-level features by spatially max-pooling. Since the process of codebook training is basically the process of finding cluster centers, we directly employ  $k$ -means clustering on all clip-level features and utilize the cluster centers as priors to initialize the codebook  $C$ . This ensures that the codebook is composed of valid latent codes.

**Joint Projection.** Unlike previous methods, which optimize codebook vectors directly, our moment codebook involves training a projector  $P(\cdot)$ . It is implemented as a simple linear layer to explore correlations between different codebook vectors, aligning with the temporal correlations of video clip features. Specifically, we replace the original codebook  $C$  with the projected codebook  $C' = P(C)$ .

### 3.5. The Architecture of MQVTG

Based on the above discussion, we propose the Moment-Quantization based Video Temporal Grounding method (MQVTG) as illustrated in Fig 3, which can be divided into

two architectures: encoder-only and encoder-decoder. Regardless of the architecture, our quantization method adds only minimal parameters during training (*i.e.*, the codebook parameters) and incurs no extra cost during inference.

**Encoder-Only.** Following previous methods [17, 21], we use the pre-trained CLIP model as the spatial encoder and textual encoder. Then, to obtain the semantic-aware video representations  $z_t$ , we utilize a lightweight recurrent structure [17] as the temporal encoder, mainly consisting of cross and self-attention blocks to integrate multi-layer CLIP features. As discussed in Sec. 3.3 and Sec. 3.4,  $z_t$  is sent to the moment codebook module to learn the discriminative information, supervised by Eq. 2 and Eq. 3. Finally, following the simple design from [15], we adopt three heads on continuous feature  $z_t$  to separately predict the classification confidence score for each frame, the boundary displacements for start-end timestamps, and the saliency score. Without further specification, we employ the encoder-only architecture for the following experiments.

**Encoder-Decoder.** Since our moment quantization is easy to implement, it can be easily integrated into the general encoder-decoder (DETR) architecture. As shown in Fig. 3, replacing our temporal encoder with a transformer encoder and prediction heads with a transformer decoder forms the existing general DETR architecture for VTG [21, 27, 36]. The moment codebook remains positioned between the encoder and decoder, adding discriminative information to video features. The experiment in Sec. 4.4 demonstrates the generalizability of our method.

### 3.6. Training Objectives

Our proposed MQVTG operates with losses for moment retrieval and highlight detection. Briefly, we employ L1 and

Method	Charades-STA				TACoS				Ego4D-NLQ			
	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU	R1@0.3	R1@0.5	R1@0.7	mIoU
2D-TAN [41] <i>AAAI'20</i>	58.76	46.02	27.50	41.25	40.01	27.99	12.92	27.22	4.33	1.83	0.60	3.39
VSLNet [40] <i>ACL'20</i>	60.30	42.69	24.14	41.58	35.54	23.54	13.15	24.99	4.54	2.40	1.01	3.54
M-DETR [13] <i>NeurIPS'21</i>	65.83	52.07	30.59	45.54	37.97	24.67	11.97	25.49	4.34	1.81	0.65	3.53
UniVTG [15] <i>ICCV'23</i>	70.81	58.01	35.65	50.10	<u>51.44</u>	34.97	17.35	33.60	<b>7.28</b>	3.95	1.32	4.91
R <sup>2</sup> -Tuning [17] <i>ECCV'24</i>	70.91	<b>59.78</b>	<u>37.02</u>	<u>50.86</u>	49.71	<u>38.72</u>	<u>25.12</u>	35.92	<u>7.20</u>	<u>4.49</u>	<u>2.12</u>	<u>4.94</u>
LLMEPET [11] <i>ACMMM'24</i>	<u>70.91</u>	—	36.49	50.25	<b>52.73</b>	—	22.78	<b>36.55</b>	—	—	—	—
<b>MQVTG (Ours)</b>	<b>70.97</b>	<u>58.84</u>	<b>38.84</b>	<b>51.10</b>	49.91	<b>39.09</b>	<b>25.82</b>	<u>36.15</u>	7.18	<b>4.78</b>	<b>2.35</b>	<b>5.08</b>

Table 2. Moment retrieval results on Charades-STA, TACoS and Ego4D-NLQ.

Method	Dog	Gym	Par.	Ska.	Ski.	Sur.	Avg.	Method	VT	VU	GA	MS	PK	PR	FM	BK	BT	DS	Avg.
UMT [16]	65.9	75.2	<b>81.6</b>	71.8	72.3	82.7	74.9	UMT [16]	87.5	81.5	88.2	78.8	81.4	87.0	76.0	86.9	84.4	<u>79.6</u>	83.1
QD-DETR [21]	72.2	<u>77.4</u>	71.0	72.7	72.8	80.6	74.4	QD-DETR [21]	<b>88.2</b>	87.4	85.6	85.0	85.8	86.9	76.4	91.3	89.2	73.7	85.0
UniVTG [15]	71.8	76.5	73.9	73.3	73.2	82.2	75.2	UniVTG [15]	83.9	85.1	89.0	80.1	84.6	81.4	70.9	91.7	73.5	69.3	81.0
CG-DETR [20]	<u>76.3</u>	<u>76.1</u>	70.0	<u>76.0</u>	<u>75.1</u>	81.9	75.9	CG-DETR [20]	86.9	<u>88.8</u>	<b>94.8</b>	<b>87.7</b>	86.7	<u>89.6</u>	74.8	<u>93.3</u>	89.2	75.9	<u>86.8</u>
UVCOM [34]	73.8	77.1	75.7	75.3	74.0	82.7	<u>76.4</u>	UVCOM [34]	87.6	<b>91.6</b>	91.4	<u>86.7</u>	<u>86.9</u>	86.9	76.9	92.3	87.4	75.6	86.3
R <sup>2</sup> -Tuning [17]	75.6	73.5	73.0	74.6	74.8	<u>84.8</u>	76.1	R <sup>2</sup> -Tuning [17]	85.0	85.9	91.0	81.7	<b>88.8</b>	87.4	<u>78.1</u>	89.2	<u>90.3</u>	74.7	85.2
<b>MQVTG</b>	<b>76.6</b>	<b>79.7</b>	<u>75.8</u>	<b>77.6</b>	<b>76.4</b>	<b>84.9</b>	<b>78.5</b>	<b>MQVTG</b>	<u>87.7</u>	<b>91.6</b>	<u>92.3</u>	85.2	85.7	<b>91.3</b>	<b>78.5</b>	<b>96.5</b>	<b>90.6</b>	<b>82.9</b>	<b>88.2</b>

Table 3. Highlight detection results on Youtube HL.

Table 4. Highlight detection results on TVSum.

focal objectives for moment retrieval loss  $\mathcal{L}_{mr}$  and use intra-video contrastive loss [15] as highlight detection loss  $\mathcal{L}_{hd}$ . As discussed in Sec. 3.3, we use  $\mathcal{L}_{mq} = \mathcal{L}_{cb} + \lambda_{cmt}\mathcal{L}_{cmt}$  as the supervision of moment quantization. We also adopt InfoNCE loss [23] as the alignment loss  $\mathcal{L}_{align}$  to calculate the video-level and layer-wise constraints of temporal encoder  $E_t$  [17]. The overall objectives can be formulated as:

$$\mathcal{L}_{overall} = \mathcal{L}_{mr} + \lambda_{hd}\mathcal{L}_{hd} + \lambda_{mq}\mathcal{L}_{mq} + \lambda_{align}\mathcal{L}_{align}, \quad (4)$$

where  $\lambda_*$  are the balancing parameters. Refer to the supplemental material for details about the training objectives.

## 4. Experiments

### 4.1. Datasets and Metrics

**Datasets.** We evaluate the proposed method on six popular video temporal grounding benchmarks, including QVHighlights [13], Charades-STA [6], TACoS [25], Ego4D-NLQ [7], YouTube Highlights [28] and TVSum [26]. Details of each dataset are included in supplemental material.

**Metrics.** Following [15, 20], we measure the performance of our model by the same criteria for QVHighlights, Charades-STA, TACoS, Ego4D-NLQ, YouTube Highlights and TVSum. For details of the metrics corresponding to datasets, please see the supplemental material.

### 4.2. Implementation Details

Following previous methods [17, 21], we employ the pre-trained CLIP model [24] as the spatial encoder and textual encoder. For the encoder-only architecture, we utilize a

lightweight recurrent structure [17] as the temporal encoder. Without further specification, we employ the encoder-only architecture for the following experiments. We set the embedding dimension  $d$  to 256. The size of the moment codebook  $K$  is set to 1024.

### 4.3. Performance Comparison

**QVHighlights.** As shown in Tab. 1, we first compare our method to previous methods on QVHighlights. Compared to previous methods that focus on learning continuous features, our method achieves the best performance on most metrics. Interestingly, we observe that the performance improvement on highlight detection is less pronounced than on moment retrieval. We attribute this to the difficulty of moment quantization in simultaneously meeting the demands of both moment retrieval (focusing on local relationships) and highlight detection (focusing on global information).

**Charades-STA, TACoS & Ego4D-NLQ.** We report the results of three moment retrieval datasets in Tab. 2. MQVTG still works better than all previous methods. However, we observe that while our results are notably superior on QVHighlights, the margin is relatively small on these three datasets. We attribute this to low codebook utilization, which is below 10%—a common issue in previous works [30, 42]. Additionally, the scene similarity across these datasets requires model to capture fine-grained details. Therefore, codewords intended to provide fine-grained information are not activated, limiting to capture details.

**YouTube Highlights & TVSum.** The results of highlight detection on YouTube Highlights and TVSum are reported

Method	R1		mAP		
	@0.5	@0.7	@0.5	@0.75	Avg.
QD-DETR <sup>‡</sup> [21]	62.58	46.45	62.84	42.44	41.86
+ <i>Moment Quantization</i>	63.48	50.00	63.28	44.35	43.63
TR-DETR <sup>‡</sup> [27]	67.16	51.10	66.73	45.37	44.89
+ <i>Moment Quantization</i>	67.68	52.00	66.97	46.24	45.41
Taskweave <sup>‡</sup> [36]	64.19	50.77	64.60	46.19	45.30
+ <i>Moment Quantization</i>	65.16	51.61	65.05	47.21	45.86

Table 5. Generalizability evaluation on QVHighlights *val* split. ‡ indicates the model is reproduced by the official codebase.

Method	R1		mAP	
	@0.5	@0.7	@0.5	Avg.
Baseline (w/o quantization)	65.35	49.42	66.99	45.63
+ QATM	66.37	51.11	67.43	47.02
+ QATM + SQ	66.52	51.23	68.18	47.54
+ QATM + SQ + MC (MQVTG)	<b>67.94</b>	<b>53.03</b>	<b>68.54</b>	<b>48.81</b>

Table 6. Ablation study of components on QVHighlights. Baseline denotes the model without quantization. ‘QATM’ denotes quantization after temporal modeling. ‘SQ’ denotes soft quantization with continuous features. ‘MC’ denotes moment codebook.

in Tab. 3 and Tab. 4. From an overall perspective, our method gains significant improvements, outperforming the state-of-the-art methods by 2.1% and 1.4%, respectively.

#### 4.4. Generalizability Evaluation

As discussed in Sec. 3.5, our moment quantization can be easily integrated into the general encoder-decoder (DETR) architecture. To investigate this property, we conduct experiments by adopting moment quantization on three DETR models. As shown in Tab. 5, our method shows strong adaptability with existing frameworks, demonstrating the effectiveness of moment quantization.

#### 4.5. Ablation Study

**Main Ablation.** We first investigate the effectiveness of each component discussed in Sec. 3.3 and 3.4. As shown in Tab. 6, we report the impact according to quantization after temporal modeling, soft quantization, and moment codebook. The results demonstrate that each component contributes significantly to overall performance.

**Quantization Method.** In Tab. 7, we explore the effectiveness of three different quantization methods discussed in Fig. 2, including image, clip and moment quantization. Image quantization performs patch-level quantization before spatial pooling. Three quantization methods significantly improve performance compared with the baseline results in Tab. 6, indicating that discrete learning from vector quantization benefits the VTG task. Our moment quantization

Method	Changes	R1		mAP	
		@0.5	@0.7	@0.5	Avg.
Quantization Method	Image	67.16	51.03	67.46	46.55
	Clip	66.84	51.61	67.32	46.93
	Moment	<b>67.94</b>	<b>53.03</b>	<b>68.54</b>	<b>48.81</b>
D/C Fusion Method	Hard	67.21	50.90	67.85	47.46
	Concat	67.35	49.74	68.20	47.60
	Add	<b>68.39</b>	50.84	68.48	47.88
	Soft	67.94	<b>53.03</b>	<b>68.54</b>	<b>48.81</b>

Table 7. Ablation study on the quantization method and discrete/continuous features fusion method.

Method	Changes	R1		mAP	
		@0.5	@0.7	@0.5	Avg.
Codebook Initialization	Random	67.61	49.55	68.91	46.89
	Selection	<b>68.00</b>	51.42	<b>69.22</b>	47.59
	Clustering	67.94	<b>53.03</b>	68.54	<b>48.81</b>
Codebook Projection	Frozen	66.17	50.26	66.62	46.92
	Basic	67.48	50.97	68.33	47.86
	Projected	<b>67.94</b>	<b>53.03</b>	<b>68.54</b>	<b>48.81</b>
Codebook Size	512	<b>68.12</b>	52.82	<b>68.75</b>	48.77
	1024	67.94	<b>53.03</b>	68.54	<b>48.81</b>
	2048	67.91	52.89	68.41	48.68
Codebook Transferability	Ch→QV	67.54	52.91	<b>69.19</b>	48.45
	TA→QV	67.72	52.86	68.33	48.63
	QV→QV	<b>67.94</b>	<b>53.03</b>	68.54	<b>48.81</b>

Table 8. Ablation study on moment codebook, including initialization, projection, size and transferability. ‘QV’, ‘Ch’, and ‘TA’ are short for QVHighlights, Charades, and TACoS, respectively.

achieves optimal performance through meticulous designs.

**Discrete/Continuous Features Fusion Method.** In Tab. 7, besides hard quantization (directly using discrete features), we design two different fusion modes for continuous and discrete features: ‘‘Add’’ means directly adding them and ‘‘Concat’’ means the concatenation operation. The results show that incorporating discrete features in any manner fails to achieve optimal performance.

**Codebook Initialization.** As shown in Tab. 8, we explore different codebook initialization methods, including random initialization and random selection, where the initial features are randomly chosen from all clip-level features in the dataset. Our *k*-means method generates representative features for initialization, resulting in a better codebook.

**Codebook Projection.** We provide a comparison between the projected, basic and frozen codebook in Tab. 8, where ‘frozen’ means remaining fixed during training. The results show that the learned codebook, especially incorporating the projector, significantly improves performance.

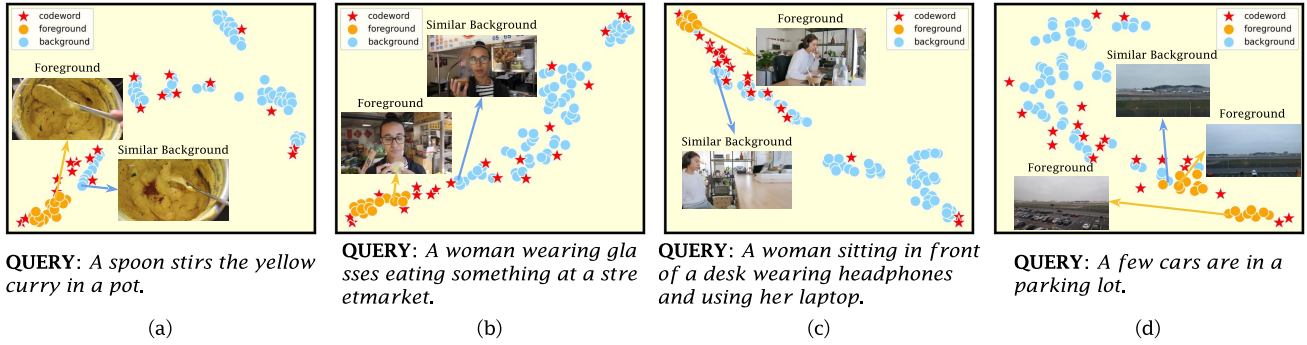


Figure 4. Visualization of effective codebook vectors, foreground and background features in the latent space. (a), (b) and (c) are three successful cases, and (d) is a failure case. For better understanding, we provide both foreground and similar background frames for each example. With codebook assistance, our method performs strong foreground aggregation and fore/background separation across scenarios.

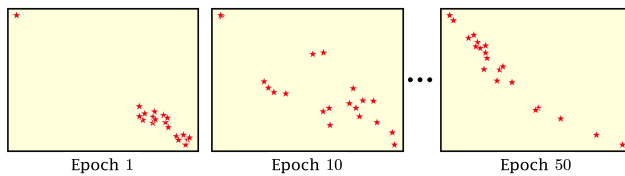


Figure 5. Evolution of effective codebook vectors during training.

**Codebook Size.** In Tab. 8, we explore the impact of codebook size. In previous image quantization methods, the size is typically set to 1024, so we adopt this setting. The results show that our method is not sensitive to codebook size.

**Codebook Transferability.** In Tab. 8, we examine the transferability of our codebook by initializing it with one dataset and training on another. The results indicate that our codebook is not dependent on the specific dataset distribution, demonstrating the robustness of our codebook.

#### 4.6. Codebook Analysis

**High-level Clustering.** In Fig. 4, we provide four examples of the distribution of effective codebook vectors and video features. It can be observed that moment quantization performs higher-level clustering with fewer codebook vectors, directly grouping foreground and background features instead of specific actions or scenes, likely due to lower codebook utilization. As a result, most moments share same codebook vectors. The three successful cases (a), (b), and (c) demonstrate our method’s strong foreground aggregation and fore/background separation across different scenarios. In the failure case (d), due to low codebook utilization, it performs poorly in clustering foreground and background that require fine-grained information for discrimination.

**Necessity of Soft Quantization.** In Fig. 4, our method uses same limited codewords for moments with different semantics. It means that with hard quantization, all moments from different videos would be represented by same codewords, clearly leading to the loss of distinctive information.

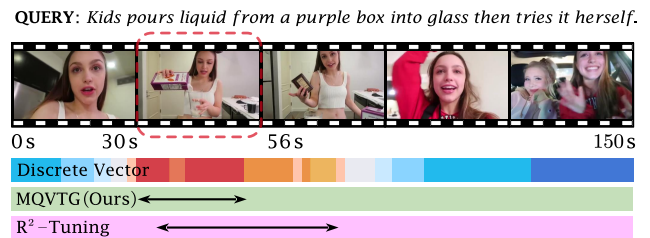


Figure 6. Qualitative result on the QVHighlights dataset.

**Evolution of Codebook.** In Fig. 5, we visualize the evolution of the distribution of effective codewords during training. As expected, the codewords gradually disperse, allowing them to represent different moments.

#### 4.7. Qualitative Result

As shown in Fig. 6, we visualize a qualitative result of MQVTG on the QVHighlights dataset. The discrimination introduced by moment quantization allows our method to localize timestamps of the moment precisely. Comparatively, without the discriminative information,  $R^2$ -Tuning [17] struggles to handle moments in similar scenes.

### 5. Conclusion

In this paper, we propose a Moment-Quantization based Video Temporal Grounding method (MQVTG) to enhance the discrimination between relevant and irrelevant moments. To adapt vector quantization from images to videos, we introduce two progressive implementations, clip quantization and moment quantization, both capable of quantizing video features to capture discriminative information. Clip quantization simply aligns with image quantization, while moment quantization makes significant improvements to meet the cross-clip nature and visual diversity of video moments. Extensive experiments and analysis demonstrate our method effectively groups relevant features and separates irrelevant ones, achieving state-of-the-art performance.

## Acknowledgements

This work was supported in part by the National Key Research and Development Project under Grant 2024YFB4708100, National Natural Science Foundation of China under Grants 62088102, U24A20325 and 12326608, and Key Research and Development Plan of Shaanxi Province under Grant 2024PT-ZCK-80.

## References

- [1] Evlampios Apostolidis, Eleni Adamantidou, Alexandros I Metsai, Vasileios Mezaris, and Ioannis Patras. Video summarization using deep neural networks: A survey. *Proceedings of the IEEE*, 109(11):1838–1863, 2021. 1
- [2] Taivanbat Badamdorj, Mrigank Roachan, Yang Wang, and Li Cheng. Joint visual and audio learning for video highlight detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8127–8137, 2021. 2
- [3] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 3
- [4] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, pages 1–2. Prague, 2004. 3
- [5] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 1, 3, 4
- [6] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*, pages 5267–5275, 2017. 1, 2, 6
- [7] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 6
- [8] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10696–10706, 2022. 3
- [9] Fa-Ting Hong, Xuanteng Huang, Wei-Hong Li, and Wei-Shi Zheng. Mini-net: Multiple instance ranking network for video highlight detection. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 345–360. Springer, 2020. 2
- [10] Jinhyun Jang, Jungin Park, Jin Kim, Hyeongjun Kwon, and Kwanghoon Sohn. Knowing where to focus: Event-aware transformer for video grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13846–13856, 2023. 2
- [11] Yiyang Jiang, Wengyu Zhang, Xulu Zhang, Xiaoyong Wei, Chang Wen Chen, and Qing Li. Prior knowledge integration via llm encoding and pseudo event regulation for video moment retrieval. In *ACM Multimedia 2024*, 2024. 1, 5, 6
- [12] Pilhyeon Lee and Hyeran Byun. Bam-detr: Boundary-aligned moment detection transformer for temporal sentence grounding in videos. *arXiv preprint arXiv:2312.00083*, 2023. 5
- [13] Jie Lei, Tamara L Berg, and Mohit Bansal. Detecting moments and highlights in videos via natural language queries. *Advances in Neural Information Processing Systems*, 34: 11846–11858, 2021. 1, 2, 5, 6
- [14] Wenrui Li, Xiaopeng Hong, and Xiaopeng Fan. Spikemba: Multi-modal spiking saliency mamba for temporal video grounding. *arXiv preprint arXiv:2404.01174*, 2024. 5
- [15] Kevin Qinghong Lin, Pengchuan Zhang, Joya Chen, Shraman Pramanick, Difei Gao, Alex Jinpeng Wang, Rui Yan, and Mike Zheng Shou. Univtg: Towards unified video-language temporal grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2794–2804, 2023. 1, 5, 6
- [16] Ye Liu, Siyuan Li, Yang Wu, Chang-Wen Chen, Ying Shan, and Xiaohu Qie. Umt: Unified multi-modal transformers for joint video moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3042–3051, 2022. 1, 6
- [17] Ye Liu, Jixuan He, Wanhua Li, Junsik Kim, Donglai Wei, Hanspeter Pfister, and Chang Wen Chen. R2-tuning: Efficient image-to-video transfer learning for video temporal grounding. In *European Conference on Computer Vision*, pages 421–438. Springer, 2024. 2, 5, 6, 8
- [18] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete representations strengthen vision transformer robustness. *arXiv preprint arXiv:2111.10493*, 2021. 3
- [19] Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. Generation-augmented retrieval for open-domain question answering. *arXiv preprint arXiv:2009.08553*, 2020. 1
- [20] WonJun Moon, Sangeek Hyun, SuBeen Lee, and Jae-Pil Heo. Correlation-guided query-dependency calibration in video representation learning for temporal grounding. *arXiv preprint arXiv:2311.08835*, 2023. 5, 6
- [21] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. Query-dependent video representation for moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23023–23033, 2023. 2, 5, 6, 7
- [22] Nasser M Nasrabadi and Robert A King. Image coding using vector quantization: A review. *IEEE Transactions on communications*, 36(8):957–971, 1988. 3
- [23] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3, 6
- [24] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,

- Amanda Askeff, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 6
- [25] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzell, Stefan Thater, Bernt Schiele, and Manfred Pinkal. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics*, 1:25–36, 2013. 6
- [26] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187, 2015. 6
- [27] Hao Sun, Mingyao Zhou, Wenjing Chen, and Wei Xie. Tdetr: Task-reciprocal transformer for joint moment retrieval and highlight detection. *arXiv preprint arXiv:2401.02309*, 2024. 1, 2, 5, 7
- [28] Min Sun, Ali Farhadi, and Steve Seitz. Ranking domain-specific highlights by analyzing edited videos. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part I 13*, pages 787–802. Springer, 2014. 2, 6
- [29] Xiaolong Sun, Liushuai Shi, Le Wang, Sanping Zhou, Kun Xia, Yabing Wang, and Gang Hua. Diversifying query: Region-guided transformer for temporal sentence grounding. *arXiv preprint arXiv:2406.00143*, 2024. 5
- [30] Yuhta Takida, Takashi Shibuya, WeiHsiang Liao, Chieh-Hsin Lai, Junki Ohmura, Toshimitsu Uesaka, Naoki Murata, Shusuke Takahashi, Toshiyuki Kumakura, and Yuki Mitsu-fuji. Sq-vae: Variational bayes on discrete representation with self-annealed stochastic quantization. *arXiv preprint arXiv:2205.07547*, 2022. 6
- [31] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 3, 4
- [32] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015. 1
- [33] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015. 1
- [34] Yicheng Xiao, Zhuoyan Luo, Yong Liu, Yue Ma, Hengwei Bian, Yatai Ji, Yujiu Yang, and Xiu Li. Bridging the gap: A unified video comprehension framework for moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18709–18719, 2024. 1, 5, 6
- [35] Bo Xiong, Yannis Kalantidis, Deepti Ghadiyaram, and Kristen Grauman. Less is more: Learning highlight detection from video duration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1258–1267, 2019. 1
- [36] Jin Yang, Ping Wei, Huan Li, and Ziyang Ren. Task-driven exploration: Decoupling and inter-task feedback for joint moment retrieval and highlight detection. *arXiv preprint arXiv:2404.09263*, 2024. 1, 5, 7
- [37] Zhou Yang, Weisheng Dong, Xin Li, Mengluan Huang, Yulin Sun, and Guangming Shi. Vector quantization with self-attention for quality-independent representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24438–24448, 2023. 3, 4
- [38] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021. 1, 3, 4
- [39] Baoquan Zhang, Huaibin Wang, Chuyao Luo, Xutao Li, Guotao Liang, Yunming Ye, Xiaochen Qi, and Yao He. Codebook transfer with part-of-speech for vector-quantized image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7757–7766, 2024. 1, 3
- [40] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. Span-based localizing network for natural language video localization. *arXiv preprint arXiv:2004.13931*, 2020. 6
- [41] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. Learning 2d temporal adjacent networks for moment localization with natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12870–12877, 2020. 6
- [42] Chuanxia Zheng and Andrea Vedaldi. Online clustered codebook. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22798–22807, 2023. 6