VideoMiner: Iteratively Grounding Key Frames of Hour-Long Videos via Tree-based Group Relative Policy Optimization

Xinye Cao[1*] Hongcan Guo[1*] Jiawen Qian[1*] Guoshun Nan[1†] Chao Wang[1]
Yuqi Pan[1] Tianhao Hou[1] Xiaojuan Wang[1] Yutong Gao[2]

[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]Minzu University of China, Beijing, China
{caoxinye, ai.guohc, qjwww, nanguo2021, wangchao0317, panyuqi2022, datoucai, wj2718}@bupt.edu.cn, ytgao92@muc.edu.cn

A. Method

In this section, we first introduce the overall process of our proposed VideoMiner. Subsequently, we provide a detailed description of the proposed T-GRPO procedure. Through this structure, we aim to offer a comprehensive understanding of the VideoMiner's functionality and the operational mechanisms of T-GRPO.

A.1. Workflow of the proposed VideoMiner

The proposed VideoMiner consists of three main components: scene segmentation and caption generation, T-GRPO-based tree exploration, and reasoning using a large language model (LLM). The input long video is temporally segmented into events, which are then processed by a vision language model (VLM) to generate captions based on a given question. We perform clustering on the generated captions, treating each cluster as a tree node. The policy model within T-GRPO determines whether a node should continue to expand. If further expansion is deemed necessary, the node undergoes an iterative process that includes segmentation, caption generation, and clustering to create new child nodes. This iterative process continues until the policy model identifies all key frames. Finally, the captions of the key frames, along with the original question, are input into the VLM to perform reasoning and provide the final answer.

A.1.1. Event Segmentation

Hour-long videos contain a vast amount of redundant information that is unrelated to the given question. To mitigate the complexity of long videos and form a hierarchical structure, we first apply uniform sampling and segment the video based on distinct scenes. By adaptively segmenting the video at the event level rather than using discrete frames, we effectively preserve temporal coherence, minimizing the disruption of temporal information during both the segmentation and subsequent clustering processes. We formulate the complete process below.

A long video, after uniform sampling into N frames, can be represented as a set $\mathcal{F}_i = \{f_1, \dots, f_t, \dots, f_N\}$. In this process, each frame f_t is represented by a normalized grayscale histogram, capturing the distribution of intensity levels within the image and providing essential information about the content. The calculation of the normalized grayscale histogram for frame f_t is given by:

$$H_t(k) = \frac{1}{WH} \sum_{i=1}^{W} \sum_{j=1}^{H} \delta\left(\text{gray}(f_t(i,j)) - k\right), \quad (1)$$

where $H_t(k)$ denotes the normalized histogram value at grayscale level k, $W \times H$ represents the resolution of the image (its width W and height H), and $\text{gray}(f_t(i,j))$ signifies the grayscale intensity at coordinate (i,j) in frame t. The Kronecker delta function $\delta(x)$ is defined as:

$$\delta(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise} \end{cases}. \quad (2)$$

This means the calculated grayscale value contributes to the histogram only when it matches the current grayscale level k. To quantify the similarity between frames, we employ the Bhattacharyya distance between consecutive histogram distributions, which measures the similarity of two probability distributions. The constructed similarity sequence can be expressed as:

$$D_i = -\ln\left(\sum_{k=0}^{255} \sqrt{H_i(k) H_{i+1}(k)}\right), \quad (3)$$

*Equal contribution
†Corresponding author

where $H_i(k)$ and $H_{i+1}(k)$ represent the normalized grayscale histograms of frames i and $i+1$, respectively. By calculating the Bhattacharyya distance D_i between adjacent frames, we can identify changes in video content, providing a basis for subsequent scene segmentation. The segmentation threshold τ is determined by selecting the top $K-1$ largest change points from the calculated distances D_i, which reflect significant differences between scenes. Let D_{sorted} be the sorted array of distances, then the segmentation points p_m can be expressed as:

$$p_m = \text{argmax}(D_{\text{sorted}}[m]), \quad m = 1, \ldots, K-1. \tag{4}$$

Once the corresponding segmentation points $\{p_1, \ldots, p_{K-1}\}$ are identified, we obtain the scene partitions:

$$E_m = \begin{cases} \{f_1, \ldots, f_{p_1}\} & m = 1 \\ \{f_{p_{m-1}+1}, \ldots, f_{p_m}\} & 2 \le m \le K-1 \\ \{f_{p_{K-1}+1}, \ldots, f_N\} & m = K \end{cases}. \tag{5}$$

In this equation, E_m denotes the m-th scene, and the partitions delineate the segments of the video based on identified change points. After scene segmentation, the input long video $\mathcal{F}_i$ is effectively partitioned into K distinct scenes $E = \{E_1, \ldots, E_K\}$. This segmentation provides a crucial foundation for subsequent video analysis, processing, and understanding, enabling a better grasp of the structure and dynamics of video content. Furthermore, the effectiveness of this segmentation can be validated by analyzing the histogram distributions and their corresponding distances, ensuring that the identified scenes accurately reflect the underlying content changes in the video. This methodology enhances the capability to process long videos and facilitates a structured approach to content analysis, leading to improved insights and applications in various domains.

A.1.2. Caption Generation and Clustering

Each event consists of a continuous sequence of frames, which construct a specific video segment. To better capture specific information relevant to the user-provided question Q and improve clustering efficiency, we utilize a Visual Language Model (VLM). This model processes video content and generates relevant textual descriptions, thereby producing captions for each event. The caption for the m-th event is defined as follows:

$$\text{Caption}_m = \text{VLM}(E_m, Q), \quad m = 1, \ldots, K, \tag{6}$$

where E_m represents the sequence of frames for the m-th event, and Q is the question posed by the user. This means that the VLM takes both the event frames and the user question as inputs to generate a contextually relevant caption. The generated captions not only reflect the main content of the events but also directly relate to the user's needs, ensuring the relevance and accuracy of the information. To transform a long video into a hierarchical tree structure, it is essential to effectively cluster the events to form tree nodes. First, each textual description Caption_m is mapped to a vector representation using an embedding model. This conversion process transforms the textual information into a numerical format suitable for analysis:

$$v_m = \text{Embedding}(\text{Caption}_m), \tag{7}$$

where v_m is the vector representation of the m-th caption. The extracted embeddings form a feature matrix $V \in \mathbb{R}^{K \times d}$, where d is the dimension of the embedding vectors:

$$V = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_K \end{bmatrix}. \tag{8}$$

This feature matrix allows us to compare and analyze the similarities among different events in a higher-dimensional space. Next, we apply a density-based clustering algorithm, DBSCAN, to group the K sub-scenes into C semantic events that exhibit similar characteristics. The clustering process can be expressed as:

$$\{E_1, ..., E_K\} \xrightarrow[\epsilon,\text{minPts}]{\text{DBSCAN}} \{E'_1, ..., E'_C\}, \tag{9}$$

In this equation, ϵ denotes the neighborhood radius, defining the range of each point's neighborhood, while minPts signifies the minimum density threshold, ensuring that each cluster contains at least a specified number of points. The result is that C clusters are formed such that $C \le K$, ensuring that semantically related scenes are grouped together to form higher-level structural nodes within the tree. This clustering method not only enhances the efficiency of analysis, but also aids in achieving a clearer understanding of the structural organization of video content. By systematically organizing and categorizing events, we can better extract valuable information from the video data, thereby providing users with more precise services and experiences. This organized approach facilitates subsequent processing and analysis, ensuring that the information derived from the video is both relevant and actionable.

A.1.3. Tree Exploration

After segmentation, caption, and clustering to form tree nodes N, the policy model in our proposed T-GRPO decides which nodes can iteratively expand into new nodes. As the tree grows, the long video is decomposed into a hierarchical structure, progressing from coarse to fine, where a deeper layer of the tree contains more fine-grained information. The action of the policy model includes three states: accept, continue, and delete. Specifically, **Accept** indicates that the node contains sufficient key frames to answer the question, requiring no further exploration. **Continue** suggests that the node may be relevant to the query and should be further expanded to new nodes for deeper exploration. **Delete** signifies that the node is irrelevant to the question and can be discarded without further expansion.

As the core component, the PM policy model determines the tree growth process, which is designed based on three aspects: spatio-temporal information integration, question-oriented, and structural adaptability. Following the three design principles, the policy model takes three inputs: event captions Caption_m, the user question Q, and node depth $\text{depth}(N_i)$. The output of the policy model $\text{State}(N_i)$ can be represented as:

$$\text{State}(N_i) = \text{PM}(\text{Caption}_m, Q, \text{depth}(N_i)). \tag{10}$$

The event captions preserve the temporal continuity of the original long video, while the question-driven captions reflect spatial information. Incorporating the question ensures that the model's decision-making remains closely aligned with the user's intention. The depth of the node provides positional information within the hierarchical structure. In Section 3.2, we introduce the concept of tree auxin to regulate excessive exploration, thus enhancing the accuracy and efficiency of localization.

All nodes with the state of accept represent the selected key frames. These key frames are collected and, along with the user's question, are fed into the VLM for inference to generate the final result. This exploration of the tree structure not only enhances the efficiency of video analysis but also enables users to access content relevant to their queries rapidly. By managing the complexity of long videos through this hierarchical approach, users benefit from a more precise and efficient experience when conducting information retrieval. Ultimately, the policy model, through intelligent node management and a dynamic decision-making process, ensures the in-depth excavation and effective utilization of video content.

A.2. Tree-Based Group Relative Policy Optimization

In modern reinforcement learning, particularly within the context of Proximal Policy Optimization (PPO), value function approximation is a critical step. Traditional methods necessitate additional computational resources to accurately estimate the value function. However, T-GRPO mitigates the reliance on substantial computational resources by introducing the concept of average rewards. By leveraging the average across multiple sampled outputs, we can significantly reduce resource consumption during the training process while maintaining performance.

In the design of T-GRPO, we specifically consider the adaptation to tree structures. Tree structures offer inherent advantages in handling complex data, such as video understanding; therefore, our policy model is designed to accommodate this structure. The inputs include not only the query q but also task-relevant captions and the depth of the tree. This design enables the model to generate multiple trees, each capable of independently addressing different task nodes, thereby enhancing overall processing efficiency.

Simultaneously, the design of the reward function has been carefully considered. We decompose the original reward function into node-level and tree-level components, allowing the model to receive reward feedback at different hierarchical levels. Node-level rewards focus on the outputs of individual intermediate nodes, while tree-level rewards assess the final output of the entire tree. This hierarchical reward mechanism facilitates the model's ability to learn distinct features and decision-making processes at various levels, thereby excelling in complex tasks such as video understanding.

In the following sections, we will provide a detailed explanation of the rollout process, reward design, and loss function formulation for T-GRPO, aiding readers in comprehensively understanding the specific implementation and advantages of this innovative approach. Through these designs, we aim to achieve greater efficiency and effectiveness in the application of reinforcement learning.

Rollout Process As illustrated in Figure 3, we first employ the proposed VideoMiner process to perform a rollout, generating n distinct trees $T = \{\vec{T_1}, \ldots, \vec{T_i}, \ldots, \vec{T_n}\}$. Each tree $\vec{T_i}$ is defined as $\vec{T_i} = \{O_{i1}, \ldots, O_{ij}, \ldots, O_{iG_i}\}$, where G_i represents the total number of nodes within tree T_i. The output O_{ij} signifies the decision made by the policy model regarding the j-th node in tree T_i, specifically determining whether this node qualifies as a key frame.

From the outputs O_{ij}, we can derive several critical components: the output format f_o, which specifies the structure of the output generated by the model; the complement length l_o, which indicates the duration or extent of additional information needed to complete the output; and the action decisions a_o, which are the specific actions proposed by the model based on the analysis of the node. Each of these elements plays a vital role in the overall functionality of the model, contributing to its ability to effectively process and interpret video data. The nuanced extraction of these outputs allows for a comprehensive understanding of the model's decision-making process at each node, facilitating improved performance in tasks such as video understanding and key frame selection.

Reward Design To guide the policy model in making more structured, detailed, and accurate key-frame decisions, we design two types of rewards for each node. The first type is the node-level reward R_{node}, which evaluates the quality of individual node decisions, while the second type is the tree-level reward R_{tree}, reflecting the correctness of the final tree-level outcome. The node-level reward R_{node} is further subdivided into three components: a format reward, which is independent of the final output yet ensures structural consistency, and both length and action rewards, which directly influence the accuracy of the final result.

The format reward can be expressed as:

$$r_{\text{format}}(f_o) = \delta_{max} \cdot \mathbb{I}_{\text{max}} + \delta_{\text{corr}} \cdot \mathbb{I}_{\text{corr}}, \tag{11}$$

where $\mathbb{I}$ is an indicate function. The term corresponding to $\mathbb{I}_{\text{max}}$ indicates full compliance with the required format, resulting in a reward of δ_{max}. Conversely, the term represented by $\mathbb{I}_{\text{corr}}$ signifies partial compliance, wherein the format is still deemed sufficient for correct extraction, yielding a reward of δ_{corr}. This structure ensures that rewards are allocated based on the adherence to expected formatting standards. The completion length reward is formulated as:

$$r_{\text{length}}(l_o) = \rho \exp\left(-\frac{(l_o - l_t)^2}{2\sigma^2}\right). \tag{12}$$

In this equation, l_o represents the length of the generated response in tokens, while l_t denotes the length of the target token. The parameter σ regulates the smoothness of the reward curve, while ρ functions as a scaling factor. By modeling the reward with a Gaussian distribution, we effectively control the target token length of the response, promoting outputs that are closer to the desired length. Empirical observations indicate that an increase in response length correlates positively with overall performance. The action reward is articulated as follows:

$$\begin{aligned} r_{\text{action}}(a_o) =& \delta_d \, \mathbb{I}_{\{\text{"delete"} \in a_o\}} + \delta_a \, \mathbb{I}_{\{\text{"accept"} \in a_o\}} + \\ & \delta_c \, \mathbb{I}_{\{\text{"continue"} \in a_o\}}. \end{aligned} \tag{13}$$

In this formulation, δ_d, δ_a, and δ_c denote the reward values assigned to the actions "delete," "accept," and "continue," respectively. The reward for the "delete" action is the highest, followed by "accept," which receives a slightly lower reward, while "continue" garners the lowest reward among the three. This hierarchy reflects the importance of decisive actions in the decision-making process of the node. To further enhance the growth regulation of the tree structure, we define an auxiliary growth factor:

$$\lambda_{auxin} = \frac{\delta_d + \delta_a}{2\delta_c}. \tag{14}$$

This growth factor, inspired by the plant hormone auxin, serves to adaptively regulate tree expansion. By moderating the growth of the tree, we aim to enhance localization efficiency, ensuring that the model maintains focus on the most relevant decision paths. Among the three reward components, r_{length} and r_{action} have a direct impact on the effectiveness of the final decision. Consequently, we compute the total reward for the policy model output utilizing the following equation:

$$R_{total} = r_{format} + (r_{length} + r_{action}) \cdot R_{tree}. \tag{15}$$

This design ensures that the model incorporates both the correctness of the final decision and the control of response length and action selection. By adjusting the growth factor λ_{auxin}, the model is incentivized to favor the "accept" and "delete" actions when appropriate, ultimately enhancing efficiency while preserving decision accuracy. This comprehensive reward structure allows the model to refine its performance through continuous learning and adaptation to complex decision-making scenarios.

Loss Function. The total reward r_{ij} collected at each node (i, j) serves as the foundation for computing the group advantage A_{ij}, which quantifies how much better or worse a specific node performs relative to the average node in the entire hierarchical tree. To normalize this comparison, we standardize each node's reward by subtracting the mean reward over all nodes and dividing by the corresponding standard deviation:

$$A_{ij} = \frac{r_{ij} - \text{mean}(\{r_{11}, r_{12}, \ldots, r_{nG_n}\})}{\text{std}(\{r_{11}, r_{12}, \ldots, r_{nG_n}\})}. \tag{16}$$

Here, r_{ij} is the total reward at node (i, j), reflecting the combined node-level and tree-level feedback, and $\text{mean}(\{\cdot\})$ and $\text{std}(\{\cdot\})$ compute the empirical mean and standard deviation over all sampled node rewards $r_{11}, r_{12}, \ldots, r_{nG_n}$, where nG_n denotes the total number of nodes across n trees. Once A_{ij} is obtained, the policy model is updated by minimizing a surrogate loss function that aggregates contributions from every node while including a KL-divergence penalty to constrain policy shifts:

$$\begin{aligned} &\mathcal{J}_{T-GRPO}(\theta) \\ &\quad = \mathbb{E}[q \sim P(Q), \{o_{ij}\}_{\substack{i=1,\ldots,G \\ j=1,\ldots,N_i}} \sim \pi_{\theta_{old}}(O|q)] \\ &\left[\frac{1}{\sum_{i=1}^{n} G_i} \sum_{i=1}^{n} \sum_{j=1}^{G_i} \left(Adv_{ij} - \beta \mathbb{D}_{KL}(\pi_\theta || \pi_{ref})\right)\right], \end{aligned} \tag{17}$$

where q is the current input (e.g., the user's question), sampled from distribution $P(Q)$, and $\{o_{ij}\}$ are the node outputs (accept/continue/delete decisions) drawn according to the old policy $\pi_{\theta_{\text{old}}}$. The clipped advantage term Adv_{ij} is defined as the product of the normalized advantage and the policy probability ratio, with clipping applied to limit large updates:

$$\mathrm{Adv}_{ij} = \min\left(\frac{\pi_\theta(o_{ij} \mid q)}{\pi_{\theta_{\text{old}}}(o_{ij} \mid q)} A_{ij}, \right.$$
$$\left. \text{clip}\left(\frac{\pi_\theta(o_{ij} \mid q)}{\pi_{\theta_{\text{old}}}(o_{ij} \mid q)}, 1-\epsilon, 1+\epsilon\right) A_{ij}\right), \quad (18)$$

where clip$(\cdot)$ is a function that restricts the policy ratio within a trust region $[1-\epsilon, 1+\epsilon]$, thereby preventing overly aggressive policy updates and avoiding policy collapse. By taking the minimum of the unclipped and clipped objectives, Adv_{ij} aligns with the PPO-style surrogate objective, ensuring stable yet effective optimization. The loss function $\mathcal{J}_{T\text{-}GRPO}(\theta)$ thus jointly promotes node-level performance improvements while regularizing the policy shift across the hierarchical structure, enabling efficient and robust policy learning over complex tree-structured outputs.

B. Sample Response of the Policy Model

Sample 1: Question: What can be deduced about c's level of expertise in the task by observing the kind of adjustments made throughout the video? **Response:**<think>The node caption describes C measuring wood on the floor using a tape measure while wearing a construction uniform and focusing on the task. This indicates some level of expertise since C pays attention to detail (carefully measuring) and appears prepared (wearing proper attire). However, this single instance doesn't provide enough context to determine whether C makes frequent adjustments elsewhere in the video or completes tasks efficiently overall. Therefore, while informative, this particular scene isn't sufficient on its own to conclusively identify C's exact proficiency level across all aspects of woodworking.</think><action>continue</action>

Sample 2: Question: What is the overall purpose of c's actions in this video? how do the actions of the man contribute to this purpose? **Response:**<think>Considering the global caption and node caption, they both describe a scenario where a man is playing dominoes, which aligns closely with option E. There's no indication of building towers (A), solving puzzles (B), creating artwork (C), or teaching someone how to play dominoes specifically (D). Since the primary activity mentioned is playing dominoes, and the man's actions continue uninterrupted despite the other person looking at their phone, option E seems most fitting. Therefore, since our focus is clearly on the game of dominoes played between two individuals, we can conclude that the current node adequately addresses part of the query without requiring additional details.</think><action>stop</action>

Sample 3: Question: Describe the overall goal achieved by c throughout the video and explain how the various trowels were used in working with the cement cast. **Response:**<think>Given the global caption describes C working on a cement cast and using trowels to smooth the cement, it aligns closely with option D where C is smoothing the cement cast with a flooring trowel and a bucket trowel. However, the primary focus seems to be on demonstrating the process rather than just describing an action like smoothing alone. Since our main goal here involves understanding both the overarching objective (demonstrating the process) and the usage specifics of the trowels, we cannot solely conclude based on just one aspect. Therefore, while some elements point towards stopping due to clear indication of tool usage, the broader instructional context suggests ongoing relevance.</think><action>continue</action>

Sample 4: Question: In the process of creating the final dough piece, can you identify three main stages that c goes through, and explain how her methods and techniques change during each stage? **Response:**<think>The global caption describes the initial steps where C mixes ingredients and shapes the dough into balls. However, the question asks about three main stages including changes in methods and techniques throughout the process. Given the current node description focuses solely on mixing and shaping without detailing subsequent stages like kneading or rising, it seems insufficient to fully address the query posed. Therefore, while some relevance exists due to overlapping early actions, more comprehensive coverage is required for an accurate response.</think><action>continue</action>

Sample 5: Question: What was the primary tool used by c in the video, and how did c utilize this tool in order to create a desired outcome? **Response:**<think>Given the global caption clearly states that the primary tool used by C is a rake, and all actions described involve using a rake—leveling soil, creating a smooth surface, removing debris—it aligns perfectly with option A. Since every detail fits without needing additional information, we can conclude that no further investigation is required.</think><action>stop</action>

Sample 6: Question: Describe the overall goal achieved by c throughout the video and explain how the various trowels were used in working with the cement cast. **Response:**<think>Given the global caption describes C working on a cement cast and using trowels to smooth the cement, it aligns closely with option D where C is smoothing the cement cast with a flooring trowel and a bucket trowel. However, the primary focus seems to be on demonstrating the process rather than just describing an action like smoothing alone. Since our main goal here involves understanding both the overarching objective (demonstrating the process) and the usage specifics of the trowels, we cannot solely conclude based on just one aspect. Therefore, while some elements point towards stopping due to clear indication of tool usage, the broader instructional context suggests ongoing relevance.</think><action>continue</action>

Sample 7: Question: In the process of creating the final dough piece, can you identify three main stages that c goes through, and explain how her methods and techniques change during each stage? **Response:**<think>The global caption describes the initial steps where C mixes ingredients and shapes the dough into balls. However, the question asks about three main stages including changes in methods and techniques throughout the process. Given the current node description focuses solely on mixing and shaping without detailing subsequent stages like kneading or rising, it seems insufficient to fully address the query posed. Therefore, while some relevance exists due to overlapping early actions, more comprehensive coverage is required for an accurate response.</think><action>continue</action>

Sample 8: Question: What was the primary tool used by c in the video, and how did c utilize this tool in order to create a desired outcome? **Response:**<think>Given the global caption clearly states that the primary tool used by C is a rake, and all actions described involve using a rake—leveling soil, creating a smooth surface, removing debris—it aligns perfectly with option A. Since every detail fits without needing additional information, we can conclude that no further investigation is required.</think><action>stop</action>

Sample 9: Question: What is the primary focus of activity in the video and how does interaction between c and the child contribute to this? **Response:**<think>Given the global caption and node caption, they both describe a woman knitting a green sweater while seated on a hammock. They also mention the woman's hands working on the yarn and the child's presence adding warmth. However, none of the options provided match exactly what is described. Option E mentions knitting a scarf which differs from the sweater mentioned. Therefore, since the exact scenario isn't aligned with any given choice, we might consider stopping here because continuing would just repeat similar details without addressing the mismatched options effectively.</think><action>stop</action>

Sample 10: Question: How does c interact with the camera during the video and what might be the reason behind these adjustments? **Response:**<think>The global caption mentions that throughout the video, C adjusts the camera several times, which aligns with the question asking about how C interacts with the camera. However, the specific details provided in the node caption do not directly address why these adjustments were made (i.e., whether it was for a specific perspective or clarity). Given the options given, none explicitly state this either. Since the primary focus here seems to revolve around understanding the motivations behind adjusting the camera angles rather than describing every single action taken while doing so, we cannot conclusively say the current node fully addresses our query without additional context.</think><action>continue</action>