VENUS: A VLLM-driven Video Content Discovery System for Real Application Scenarios

Minyi Zhao[1*] **Yi Liu**[2] **Jianfeng Wen**[2]
Boshen Zhang[2] **Hailang Chang**[2] **Zhiheng Ouyang**[2]
Jie Wang[2] **Wensong He**[2] **Shuigeng Zhou**[1†]
[1]College of CS and AI, Fudan University, Shanghai, China
[2]ByteDance, Beijing, China
[1]{zhaomy20, sgzhou}@fudan.edu.cn
[2]{daniel.az, hewensong}@bytedance.com

Abstract

Video Content Discovery (VCD) is to identify the specific videos defined by a certain pre-specified text policy (or constraint), which plays a crucial role in building a healthy and high-quality Web content ecology. Currently, related works typically employ multiple classifiers or similarity-based systems to support VCD. However, these approaches are difficult to manage, lack generalization power, and suffer from low performance. To tackle these problems, this paper presents a new Vision-Language Large Model (VLLM)-driven VCD system called VENUS (the abbreviation of **V**ideo cont**EN**t **U**nder**S**tander). Concretely, we first develop an automatic policy-guided sequential annotator (APSA) to generate high-quality, VCD-specific, and reasoning-equipped instruct-tuning data for model training, then extend the VLLM inference to support VCD better. Following that, we construct a real VCD test set called VCD-Bench, which includes a total of 13 policies and 57K videos. Furthermore, to evaluate its practical efficacy, we deploy VENUS in three different simulation scenarios. Extensive experiments on both the VCD-Bench and public evaluation datasets for various VCD-related tasks demonstrate the superiority of VENUS over existing baselines.

1 Introduction

With the popularity of the Web and its applications, an increasing number of video streaming services are deployed on the web, rapidly accumulating more and more videos. These videos, while enriching a vibrant web community ecology, also pose challenges to various video content tasks (Jiang et al., 2019). Among these tasks, video content discovery (Cao et al., 2016; Helff et al., 2024), which aims to distinguish relevant or irrelevant videos according to a certain pre-specified text policy, serves as a fundamental technique for the construction of a healthy and compliant community. As shown in Fig. 1(a), given a video and a policy specifying drug-related content, a VCD system first understands the policy, then analyzes the content, finally decides whether any videos match the policy via answering "Yes/No" so that subsequent operations (*i.e.,* reject the video or distribute it to users) can be conducted to avoid harmful video content poisoning the community and distribute high-value healthy contents. Compared with typical classification and retrieval tasks, in reality VCD has the following three characteristics: 1) VCD should be able to handle massive amounts of streaming videos, while focusing on understanding related content and having few-shot/zero-shot learning capabilities, which are lacking in traditional tasks. 2) The policy (*a.k.a* query/prompt/text input) for VCD covers complex requirements and definitions, and changes frequently, which is different from

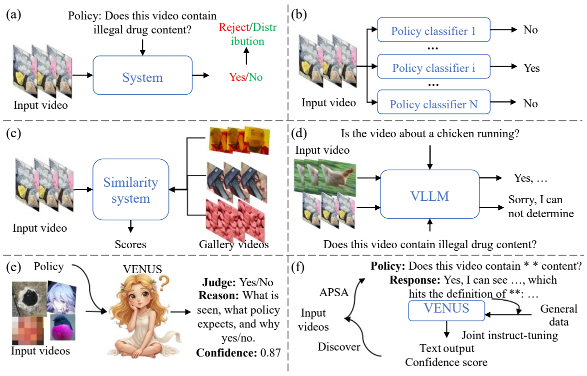


Figure 1: Illustration of VCD scheme comparison. (a) The video content discovery (VCD) task; (b) A solution based on multiple classifiers (MPC); (c) A similarity-based system (SBS); (d) A typical VLLM-based approach, utilized to do general question-answering and VCD; (e) Our VENUS that takes video and policy as input, outputs judge, reason, and confidence score to discover relevant videos; and (f) The new techniques proposed in VENUS to support VCD.

[*] This work was mainly done while the author was an intern at ByteDance.
[†] Corresponding author.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track, pages 50–64
November 4-9, 2025 ©2025 Association for Computational Linguistics

keywords and short sentences utilized in typical retrieval systems (Zhao et al., 2023). 3) VCD has to handle extremely rare visual content, which may be lacking in generic datasets. These features require a VCD system to have stronger text and video understanding ability.

Thanks to the rapid advance of artificial intelligence (AI) techniques, various deep neural network (DNN) paradigms have been proposed to support VCD. Early attempts usually rely on various DNN structures (*e.g.* CNN (Li et al., 2021), ViT (Dosovitskiy et al., 2020)) to do VCD in a video classification manner (Karpathy et al., 2014). For example, as shown in Fig. 1(b), given a policy, researchers develop a specialized model for binary classification. Nevertheless, because of the evolving nature of the Web environment, such a solution cannot adapt well to new video content and new policies. Besides, with the increase of policies, many classifiers are required, which inevitably incurs huge deployment and maintenance difficulties and costs.

To address the aforementioned drawback, recently, powered by progressive text-image alignment skills (Radford et al., 2021), similarity-based systems (SBS) have been widely used in VCD tasks. Taking Fig. 1(c) for example, researchers build a gallery that contains several representative videos (*a.k.a.* seeds) so that when a new video comes, the discovery task can be converted into the similarity measurement between the input video and the seeds. However, the performance of SBS is highly impacted by the seeds. For some rare or new policies that only have a limited number of or even no seeds, SBS shows poor performance.

Then, how to establish an effective video content discovery system? Fortunately, with the assistance of advanced Large Language Models (LLMs) (Touvron et al., 2023; Achiam et al., 2023) and support techniques (Liu et al., 2022; Peng et al., 2023), Vision Language Large Models (VLLMs) (Liu et al., 2023a, 2024b) can offer us a new solution, as shown in Fig. 1(d). The core idea of this paradigm is to utilize VLLMs' strong understanding ability for content discovery. However, since the training data of typical VLLMs are always common web content, these off-the-shelf models cannot effectively deal with long-tailed and rare-to-see videos (*e.g.* drugs, terror, self-mutilation etc.) and frequently changing policies, as also illustrated in Fig. 1(d). In addition, the textual output (*i.e.,* "Yes/No") can not support threshold adjustment well. This situation indicates that it is urgent to build powerful VLLM-driven VCD systems.

In this paper, we present a new VLLM-driven system for VCD, namely VENUS (the abbreviation of **V**ideo cont**EN**t **U**nder**S**tander). Fig. 1(e) illustrates the rationale behind VENUS. Given a video and a policy, VENUS is required to provide not only an answer of "Yes/No" but also (1) the understanding process of what is seen, what is expected by the policy, thus making the final decision, and (2) the confidence score of the decision for downstream applications. To this end, as shown in Fig. 1(f), on the one hand, VENUS proposes a novel automatic policy-guided sequential annotator (APSA) to generate high-quality instruct-tuning data for model training. Compared with typical data engines (Chen et al., 2024; Liu et al., 2024b) used by existing VLLMs. APSA not only generates reasoning-equipped question-answering (QA) data according to the policy, but also proposes a sequential cross-evaluation procedure to offer a quality evaluation phase for each generated data. On the other hand, VENUS extends the classic inference procedure to not only output the traditional text answers, but also offer confidence scores for wide usages like threshold adjustment and various downstream applications. In addition, as existing benchmarks either consider only general visual question-answering ability or exclude popular online live streams and short videos, which makes them unsuitable to evaluate VCD systems. Ergo, we construct a new, real-world, large-scale VCD-dedicated benchmark called VCD-Bench that contains 13 text policies and 57K videos.

The contributions of this paper are summarized as follows: 1) We propose a new VLLM-driven system called VENUS for video content discovery. VENUS uses a new automatic policy-guided sequential annotator to lift model's diverse policy processing ability, and provides both text and confidence outputs for better downstream usage. 2) We construct a high-quality and real video content discovery benchmark called VCD-Bench. 3) With VCD-Bench and public VLLM evaluation datasets, we conduct extensive experiments that show the advantages of VENUS over existing SOTA approaches. 4) We deploy VENUS in 3 simulation scenarios, including a large-scale running VCD system, a zero-shot retrieval task, and a few-shot setting. Testing results show that VENUS can boost performance well in all these scenarios.

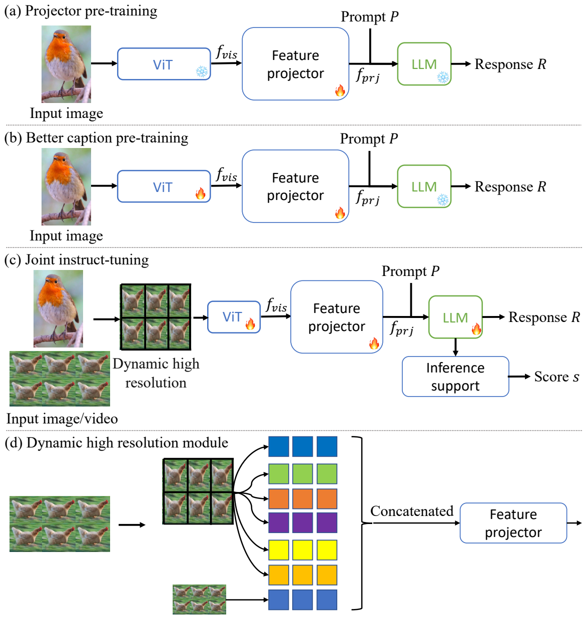


Figure 2: The architecture of VENUS (a-c) and the dynamic high-resolution module (d). The overall training procedure can be divided into three phases, including (a) Projector pre-training; (b) Caption pre-training; and (c) Joint instruct-tuning.

2 The VENUS System

In this section, we give the detailed techniques of VENUS, with focus on the architecture and the implementation of the data engine. The goal of VENUS Φ is to generate a response R and a confidence score s based on the image/video I and the policy prompt P, *i.e.*, $(R, s) = \Phi(I, P)$.

Architecture. The architecture and training procedure of VENUS are illustrated in Fig. 2. VENUS contains 5 major modules: a *ViT* Φ_{vit} used for visual feature extraction, a *feature projector* Φ_{prj} for visual feature projection, a *LLM* Φ_{llm} that is applied to generating the response based on the projected feature and the prompt P, a *dynamic high-resolution* module (Liu et al., 2023a) Φ_{dhr} for high-resolution image processing, and a *inference support* module Φ_{ism} utilized to derive score s. The response R and score s is generated as follows:

$$R = \Phi_{llm}(\Phi_{prj}(\Phi_{vit}(\Phi_{dhr}(I))), P), \quad (1)$$

$$s = \Phi_{ism}(\Phi_{llm}(\Phi_{prj}(\Phi_{vit}(\Phi_{dhr}(I))), P)). \quad (2)$$

Concretely, given an image/video I, we first adopt the visual feature encoder Φ_{vit} to obtain its preliminary visual feature f_{vis}=$\Phi_{vit}(I)$. However, this naive approach has two shortcomings. First, existing ViTs can only support fixed input sizes

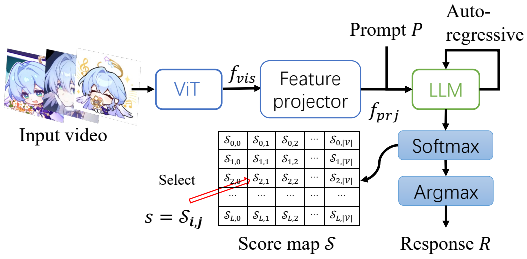


Figure 3: Illustration of inference support for VENUS.

(e.g. 384 * 384 for SigLIP (Zhai et al., 2023)), which makes the model be unable to utilize useful fine-grained pixel information. Second, capturing redundant pixel information for a long video is quite inefficient. Considering the massive amounts of high-quality image data, we propose a solution that supports both images and videos simultaneously. Accordingly, we extend the dynamic high-resolution technique (Liu et al., 2023a) Φ_{dhr} for better visual extraction, *i.e.*, $f_{vis} = \Phi_{vit}(\Phi_{dhr}(I))$. In particular, as shown in Fig. 2(d), given an input video, we sample 6 frames to form an image $I' \in \mathbb{R}^{(2\times384)\times(3\times384)}$. Then, we pre-define a 2×3 grid to crop 6 patches (384×384) from I' and extract 6 image-level features solely from these patches. We also directly downsample image I to obtain a global feature of the video. Finally, all the 7 features are concatenated, which is then fed into a multilayer perceptions projector Φ_{prj} and projected to the language domain, *i.e.*, $f_{prj} = \Phi_{prj}(f_{vis})$.

Finally, we feed the projected feature f_{prj} and the prompt P to the LLM to auto-regressively generate the response R as follows:

$$R_j = \begin{cases} \Phi_{llm}([f_{vis}, P] | < BOS >) \, (j = 0) \\ \Phi_{llm}([f_{vis}, P] | [< BOS >, R_{t<j}]) \, (j \neq 0) \end{cases} \quad (3)$$

where $[\cdot]$ and $< BOS >$ denote the concatenating operation and the *beginning of sequence* token, respectively. We also exploit the inference support module to compute the confidence score s=$\Phi_{ism}(h_t)$, where h_t is the t-th feature outputted by the LLM.

Inference Support for VCD. Given an image/video, the VCD model should generate an additional confidence score s. Then, we can adjust threshold ϵ to check whether or not the image/video hits the policy. In VENUS, we design an inference support module Φ_{ism} to obtain the corresponding score s. In particular, denote the t-th feature outputted by the LLM as h_t, we obtain the response

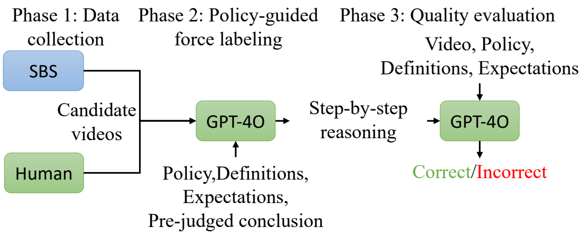


Figure 4: Illustration of our proposed automatic policy-guided sequential annotator (APSA).

R_t via two sequential functions:

$$S_t = Softmax(h_t), \\ R_t = Argmax(S_t), \tag{4}$$

where $S_t \in \mathbb{R}^{|\mathcal{V}|}$ is the t-th predicted probability distribution, $\mathcal{V}$ is the vocabulary, and $|\mathcal{V}|$ denotes the size of the vocabulary. As shown in Fig. 3, in VENUS we cache all the distributions to form the final score table $\mathcal{S} = \{S_1, S_2, .., S_L\}$, where L is the index of the <EOS> (*end of sequence*) token. Then, we can easily obtain the confidence score by parsing the response R. For example, consider a binary classification prompt, the key token in R should be "Yes" (or "No"). Let the index of the key token in the response and the vocabulary be **i** and **j**, respectively. The score s can be extracted as follows: $s = \mathcal{S}_{\mathbf{i},\mathbf{j}}$. During inference, once key tokens are generated, the system can immediately return the score instead of continuing unnecessary computations, thus saving computational resources.

Automatic Policy-guided Sequential Annotator (APSA). This section focuses on the data engine to create training data. A straightforward solution is to write some policy-specific questions and then directly use a strong VLLM (*e.g.* GPT-4O) to answer the questions. However, because of the challenge of rare policies and content in the VCD scenario, even powerful VLLMs cannot always provide correct answers. Inspired by existing similarity-based systems and forced teaching techniques, we propose a *automatic policy-guided sequential annotator* (APSA) to generate high-quality yet economic policy-specific data such that we can mix these data with public data for model training.

As shown in Fig. 4(a), ASAP consists of three phases: data collection, policy-guided force labeling, and quality evaluation. The data collection phase is to get rich candidate data. Concretely, given some policies, we try to collect a batch of videos from the video service platform that are relevant/irrelevant to these policies. To this end, we consider two data collection methods. First, we use the existing similarity-based system (SBS) to capture high-confidence videos from the platform, namely *online sweeping*. Second, we use crowdsourcing to obtain some relevant videos from medium-confidence videos. As for the second phase, we not only input the definitions and expectations of the policies but also provide a pre-judged conclusion to force the VLLM to write step-by-step reasoning — what is seen, what meets the definitions, and why is relevant/irrelevant. However, in practice, there are two major causes that may lead to low-quality data. On the one hand, due to the possibility of errors in SBS, some videos will be provided with wrong conclusions, which thus impair the annotation in the first phase. On the other hand, for the sake of VLLM's own lack of expertise of rare policies and videos, the VLLM may refute our predefined conclusions, even if they are correct. Therefore, we introduce an additional quality evaluation phase to evaluate each sample we generate. As shown in Fig. 4(a), we feed the video, the question, and the answer to the VLLM to judge whether or not the QA pair is correct. We only maintain the data passed the quality evaluation phase. Case studies are given in Appendix D.

In this way, we finally obtain around 0.7M high-quality question-answer pairs for instruct-tuning.

Model Training. Here, we explain the process of VENUS training, which can be divided into three phases: projector pre-training, better caption pre-training, and joint instruct-tuning. As shown in Fig. 2(a), the first pre-training stage is utilized to align the projector Φ_{prj} via coarse-grained image-caption pairs. We prompt the LLM with "Please describe the photo." in this stage. After preliminarily aligning the projector, we apply more fine-grained image-caption data for a fine-grained alignment, namely better caption technique (Chen et al., 2023, 2024). The prompt is also changed to "Please describe the photo in detail.". In addition, we train both the visual feature encoder Φ_{vit} and the projector Φ_{prj} to enhance the fine-grained encoding performance. Finally, we unlock all the modules in VENUS and apply diverse and high-quality instruct-tuning data to boost VENUS's prompt-following ability. In the implementation, we use commercially-allowed data to develop the model — LCS-558K (Liu et al., 2024b) for pre-training, 1.2M data for better caption, and a total of 2.2M public + APSA data for instruct-tuning.

3 The VCD-Bench

Data Collection. The data collection procedure can be divided into four steps. First, we consider the discovery policies. To better validate the model performance, we select a total of 13 most popular discovery queries from the platform to serve as the policies. As a test set for VCD, its quality must be high, and the proportion of relevant and irrelevant videos (*a.k.a.* density) should be close to that in the real world. However, the real density of a rare policy is too low (around 1:10000 or even lower). Building a million-level dataset is clearly not conducive to offline testing and evaluation. Ergo, we adjust the density according to its original density and a scale factor $K = 10$. After the determination of each policy's density, what we should do is grab relevant/irrelevant videos from the Web based on the density. By taking the annotation cost into consideration, we propose to apply different strategies for relevant and irrelevant videos. On the one hand, since the goal of VCD is to find relevant videos, we only sample high-quality human-labeled videos via crowdsourcing to form the relevant subset. On the other hand, when it comes to the irrelevant subset, considering its enormous quantity, we use the similarity-based system (SBS) to automatically retrieve irrelevant videos that do not hit any policies. So far, we finalize the data collection.

Distribution Analysis. Here we present the statistics of our collected VCD-Bench. First, the volume of our dataset (57,308 videos) is significantly larger than some typical VLLM benchmarks, like MME (1187 images) and POPE (1485 images). In addition, the density of our VCD-Bench is also very close to reality. For example, the average density of VCD-Bench is 1:176.4, which means that for all 176 videos, only one is relevant. This density is more consistent with that of our running system.

Dataset Evaluation Protocol. In VCD-Bench, we consider two major evaluation metrics, which are *average precision* (AP) and *recall at precision=0.5* (R@P0.5). We do not apply the regular classification accuracy score. The reason lies in that 1) the goal of VCD is to pick out relevant videos as many as possible through threshold adjustment; 2) Given a low density, the majority of videos in the test set are labeled as irrelevant. Thus, the accuracy score will be greatly affected by these irrelevant videos.

Table 1: Performance comparison between VENUS and state-of-the-arts on various discovery and generic benchmarks. The metrics for VCD are AP and R@P0.5. '-' means the corresponding method cannot do this task.

Approach	VCD		VQA			
	AP	R@P0.5	TextVQA	MME-P	SEED-Img	MM-Vet
MPC	0.450	0.389	-	-	-	-
SBS	0.313	0.285	-	-	-	-
SigLIP	0.212	0.177	-	-	-	-
CLIP	0.191	0.151	-	-	-	-
LLaVA-Guard	0.232	0.190	49.9	1425.4	20.1	28.7
LLaVA-Next	0.202	0.153	65.7	1502.9	72.2	47.3
ShareGPT4V	0.309	0.301	60.4	1567.4	69.7	37.6
LLaVA-1.5	0.360	0.315	62.6	1549.7	71.2	53.8
LLaVA-OV	0.430	0.475	**73.8**	1584.8	**75.4**	52.7
VENUS	**0.565**	**0.573**	73.2	**1618.8**	75.2	**54.8**

4 Performance Evaluation

Implementation Details. VENUS is developed on 80 NVIDIA H100 GPUs with 80GB memory. We use WarmupCosineLR with a learning rate of 1e-3, 2e-5, and 1e-5 for three different training stages, respectively. Batch sizes for these stages are set to 128, 128, and 160 accordingly. We use AdamW (Loshchilov and Hutter, 2017) as the optimizer, and all the data is learned for one epoch. We use QWEN2.5-7B[1] and SigLIP (Zhai et al., 2023) to implement the LLM and visual feature encoder.

Compared with SOTAs. Here, we evaluate the effect of VENUS on both discovery and general capabilities. Accordingly, in addition to our VCD-Bench, we also test a total of 4 general VQA benchmarks, including TextVQA (Singh et al., 2019), MME (Fu et al., 2023), SEED-Bench (Li et al., 2023b), and MM-Vet (Yu et al., 2023). We report the accumulated score for MME and accuracy for the rest. Besides, we compare VENUS with various advanced approaches, including 1) two advanced internal systems, which are Multiple Policy Classifier (MPC) and Similarity-based System (SBS), 2) two public similarity-based methods — SigLIP (Zhai et al., 2023) and CLIP (Radford et al., 2021), and some popular VLLMs, *i.e.,* LLaVA-OV (Li et al., 2024), LLaVA-Next (Liu et al., 2024a), ShareGPT4V (Chen et al., 2023), LLaVA-1.5 (Liu et al., 2024b), and the content-safefy-specific approach LLaVAGuard (Helff et al., 2024). For a fair comparison, we use the 7B version of all these VLLMs. The LLMs for these VLLMs are set as follows: QWEN2.5 for LLaVA-1.5, QWEN2 (Yang et al., 2024) for LLaVA-OV, Mistral (Jiang et al., 2023) for LLaVA-Next, and Vicuna (Zheng et al., 2024) for ShareGPT4V. To support VCD, we install the inference technique il-

[1]https://qwenlm.github.io/blog/qwen2.5-llm/

lustrated in Sec. 2 to get the confidence score. Note that SBS, SigLIP, and CLIP require a seed gallery, we use each relevant video in the test set to serve as a high-quality seed to search for the other videos. All the experimental results are presented in Tab. 1.

We first check the VCD performance. From Tab. 1 we can see that the best VCD performance is achieved by VENUS. This indicates that VENUS is strong enough to beat the internal state-of-the-art MPC, the public VLLM LLaVA-OV, and the content-specific VLLM LLaVAGuard. Besides, MPC is a system with 13 different classifiers while VENUS has only a single model. Therefore, compared with MPC, VENUS is much easier to manage. Thus, we demonstrate VENUS' high discovery performance.

Then, we check the general performance. Obviously, the general performance of VENUS is comparable to that of LLaVA-OV. In particular, VENUS outperforms LLaVA-OV on MME and MM-Vet. As for the rest datasets, the performance gap is very slight. As a result, jointly considering both discovery and general VQA tasks, VENUS wins 3 tasks, outperforming LLaVA-OV (winning 2 tasks). Additionally, by taking the training cost of VENUS (80 GPUs and 2.2M fine-tuning data) into consideration, compared with LLaVA-OV that uses 128 $\sim$ 256 GPUs and 4.8M fine-tuning data, the overall training cost of VENUS is much lower than that of LLaVA-OV. So far, both the VCD and general performance of VENUS are verified.

Finally, we additionally sumarize some interesting observations as follows: 1) There is a positive correlation between the VCD ability and the general VQA ability. Specifically, a strong general ability can ensure a good VCD ability (See LLaVA-OV, its VCD ability outperforms LLaVA-Next, ShareGPT4V, and LLaVA). As a result, in VENUS, we decide to jointly optimize general ability and VCD performance. 2) Even though we have utilized the relevant videos in the VCD-Bench test set to serve as the seeds, the performance results of SBS, SigLIP, and CLIP are still inferior to that of MPC and VENUS. This indicates that similarity-based approaches are inferior to some high-level understanding models in practice. Ablation studies on VENUS design are given in Appendix B.

5 Applications in Simulation Scenarios

In this section, we present 3 different simulation deployment scenarios, which are illustrated in Fig. 5.

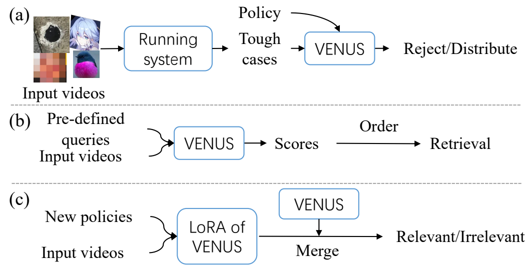


Figure 5: Three simulation deployment scenarios of VENUS. (a) Cooperating with an existing running system for VC; (b) Zero-shot application for retrieval; and (c) Fine-tuning VENUS with LoRA for urgent applications.

The first scenario is applying VENUS to a running system to exploit some harmful policies (*i.e.,* excrement, etc) that are unsuitable to be distributed on the Web, which is illustrated in Fig. 5(a). Specifically, owing to VENUS's query-per-second (QPS), we stack VENUS at the end of the VCD system to process what the current system cannot process well. Based on the results of our one-week running data, among the given policies that VENUS excels in, we can additionally discover 7.67% tough cases (around 2K videos), saving 46.3% human discovery cost.

Secondly, we apply VENUS to a zero-shot retrieval setting. As shown in Fig. 5(b), we evaluate VENUS on an internal test set that includes 10k hotels and attractions videos and 500 pre-defined hot queries. VENUS is required to retrieve the last 20 videos based on the relevance score and we find that compared with the existing system, the normalized Discounted Cumulative Gain (nDCG) is significantly boosted from 0.209 to 0.610, which leads to an online order quantity increase of 12%.

Lastly, notice that owing to the rapid expansion and evolution of the Web, new policies need to be quickly responded by VENUS. We design a fast response mechanism based on LoRA (Hu et al., 2021) to quickly provide customized temporary services to these high-demand policies via VENUS. We fine-tune the fully connected layer of each attention layer for ViT and LLM in VENUS using LoRA on specific datasets. In deployment, as shown in Fig. 5(c), only a small split of VENUS parameters are merged to support these new policies. As an example, we evaluate a new quality discerning policy and find this VENUS can boost the AP value from 0.746 to 0.894 with only 1K data, demonstrating VENUS's effectiveness in few-shot setting.

6 Conclusion

In this paper, we develop a new VLLM-driven video content discovery system called VENUS, and construct a high-quality and real VCD dataset called VCD-Bench. Extensive experiments on our constructed VCD-Bench and public general evaluation datasets, and online evaluations validate the advantages of VENUS over existing SOTAs.

7 Limitation and Future Work

As mentioned before, the main limitation of VENUS is its efficiency. Compared with MPC and SBS which has a QPS of 60, our VENUS is only 5. This indicates that VENUS is currently unable to completely replace the current system. It can only be used as a post-processing method to handle difficult samples that the current system cannot handle. In the future, we will continue to explore more useful application paradigms as well as optimize VENUS in terms of efficiency and performance: 1) Consider using techniques such as distillation (Polino et al., 2018) and inference acceleration (Pagliardini et al., 2023) to boost the efficiency of VENUS. 2) Further enhance the performance of VENUS using techniques such as Chain of Thought (CoT) (Wei et al., 2022) and Mixture of Experts (MoE) (Du et al., 2022).

8 Ethics Statement

To avoid potential societal risk, we have processed our data. The data containing personal privacy and biological information have undergone special processing (*e.g.* blurring face). The storage and destruction of data are strictly enforced in accordance with privacy laws. Therefore, all the examples provided in this article have been anonymized. In addition, we also paid crowdsourcing salaries based on local labor wages. The policies studied in this paper have also been evaluated by the legal and government relationship team.

Acknowledgements

The work was supported in part by a ByteDance Research Collaboration Project.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Juan Cao, Yongdong Zhang, Rongrong Ji, Fei Xie, and Yu Su. 2016. Web video topics discovery and structuralization with social network. *Neurocomputing*, 172:53–63.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3558–3568.

Guiming Hardy Chen, Shunian Chen, Ruifei Zhang, Junying Chen, Xiangbo Wu, Zhiyi Zhang, Zhihong Chen, Jianquan Li, Xiang Wan, and Benyou Wang. 2024. Allava: Harnessing gpt4v-synthesized data for a lite vision-language model. *arXiv preprint arXiv:2402.11684*.

Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. 2023. Sharegpt4v: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

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

Nan Du, Yanping Huang, Andrew M Dai, Simon Tong, Dmitry Lepikhin, Yuanzhong Xu, Maxim Krikun, Yanqi Zhou, Adams Wei Yu, Orhan Firat, et al. 2022. Glam: Efficient scaling of language models with mixture-of-experts. In *International Conference on Machine Learning*, pages 5547–5569. PMLR.

Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu Zheng, Ke Li, Xing Sun, et al. 2023. Mme: A comprehensive evaluation benchmark for multimodal large language models. *arXiv preprint arXiv:2306.13394*.

Chaoyou Fu, Yuhan Dai, Yondong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. 2024. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Jiaxian Guo, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Boyang Li, Dacheng Tao, and Steven Hoi. 2023. From images to textual prompts: Zero-shot visual question answering with frozen large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10867–10877.

Lukas Helff, Felix Friedrich, Manuel Brack, Kristian Kersting, and Patrick Schramowski. 2024. Llavaguard: Vlm-based safeguards for vision dataset curation and safety assessment. *arXiv preprint arXiv:2406.05113*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hongyu Hu, Jiyuan Zhang, Minyi Zhao, and Zhenbang Sun. 2023. Ciem: Contrastive instruction evaluation method for better instruction tuning. *arXiv preprint arXiv:2309.02301*.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. 2023. Llama guard: Llm-based input-output safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Yudong Jiang, Kaixu Cui, Bo Peng, and Changliang Xu. 2019. Comprehensive video understanding: Video summarization with content-based video recommender design. In *Proceedings of the IEEE/CVF international conference on computer vision workshops*, pages 0–0.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.

Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. 2023a. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. 2023b. Seed-bench: Benchmarking multimodal llms with generative comprehension. *arXiv preprint arXiv:2307.16125*.

Juncheng Li, Kaihang Pan, Zhiqi Ge, Minghe Gao, Hanwang Zhang, Wei Ji, Wenqiao Zhang, Tat-Seng Chua, Siliang Tang, and Yueting Zhuang. 2023c. Empowering vision-language models to follow interleaved vision-language instructions. *arXiv preprint arXiv:2308.04152*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023d. Blip-2: Bootstrapping language-image pretraining with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pretraining for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023e. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*.

Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, and Jun Zhou. 2021. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 33(12):6999–7019.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023a. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024a. Llava-next: Improved reasoning, ocr, and world knowledge.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, et al. 2023b. Mmbench: Is your multi-modal model an all-around player? *arXiv preprint arXiv:2307.06281*.

Shangbang Long, Xin He, and Cong Yao. 2021. Scene text detection and recognition: The deep learning era. *International Journal of Computer Vision*, 129(1):161–184.

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

Larry R Medsker, Lakhmi Jain, et al. 2001. Recurrent neural networks. *Design and Applications*, 5(64-67):2.

Vicente Ordonez, Girish Kulkarni, and Tamara Berg. 2011. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24.

Matteo Pagliardini, Daniele Paliotta, Martin Jaggi, and François Fleuret. 2023. Faster causal attention over large sequences through sparse flash attention. *arXiv preprint arXiv:2306.01160*.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Antonio Polino, Razvan Pascanu, and Dan Alistarh. 2018. Model compression via distillation and quantization. *arXiv preprint arXiv:1802.05668*.

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

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565.

Baoguang Shi, Xiang Bai, and Cong Yao. 2016. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE transactions on pattern analysis and machine intelligence*, 39(11):2298–2304.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Jiabo Ye, Anwen Hu, Haiyang Xu, Qinghao Ye, Ming Yan, Guohai Xu, Chenliang Li, Junfeng Tian, Qi Qian, Ji Zhang, et al. 2023a. Ureader: Universal ocr-free visually-situated language understanding with multimodal large language model. *arXiv preprint arXiv:2310.05126*.

Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. 2023b. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*.

Dong Yu and Lin Deng. 2016. *Automatic speech recognition*, volume 1. Springer.

Weihao Yu, Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin, Zicheng Liu, Xinchao Wang, and Lijuan Wang. 2023. Mm-vet: Evaluating large multimodal models for integrated capabilities. *arXiv preprint arXiv:2308.02490*.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986.

Minyi Zhao, Jie Wang, Zhaoyang Li, Jiyuan Zhang, Zhenbang Sun, and Shuigeng Zhou. 2024. Effectively enhancing vision language large models by prompt augmentation and caption utilization. *arXiv preprint arXiv:2409.14484*.

Minyi Zhao, Jinpeng Wang, Dongliang Liao, Yiru Wang, Huanzhong Duan, and Shuigeng Zhou. 2023.

Keyword-based diverse image retrieval by semantics-aware contrastive learning and transformer. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1262–1272.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Related Work

A.1 Video Content Discovery

Video content discovery (VCD) (Cao et al., 2016), which aims to explore videos based on a given policy, plays an important role in building a healthy Web ecology. Typical VCD approaches can be roughly divided into two categories: classifier-based and similarity-based, respectively. The classifier-based approaches usually apply various DNN models (*e.g.* CNN (Li et al., 2021), RNN (Medsker et al., 2001), ViT (Dosovitskiy et al., 2020)) for classification. Additionally, some optical character recognition (OCR) (Shi et al., 2016; Long et al., 2021) and automatic speech recognition (ASR) (Yu and Deng, 2016) techniques are applied to extract texts to assist the discovery. As for the similarity-based methods (Radford et al., 2021; Zhai et al., 2023), they usually utilize massive vision-language data (Schuhmann et al., 2022) and contrastive learning (Khosla et al., 2020) techniques to develop an alignment model so that the VCD problem is converted to a matching task. However, these methods above have obvious drawbacks. The former requires a large number of classifiers when there are many policies to handle, while the performance of the latter is heavily dependent on the video seeds. More recently, some safeguards (*i.e.,* LLama Guard (Inan et al., 2023) and LLava Guard (Helff et al., 2024)) are proposed to discover harmful contents from the Web with the help of large language model.

A.2 Vision Language Large Models

Thanks to the remarkable achievements of large language models (LLMs), many powerful vision-language large models (VLLMs) (Guo et al., 2023; Bai et al., 2023; Li et al., 2023c; Ye et al., 2023b; Li et al., 2024) have been proposed to integrate LLMs with visual encoders for various visual language understanding tasks. Generally, from the aspect of model structure, these VLLMs first use a visual encoder (*e.g.* ViT (Dosovitskiy et al., 2020), CLIP (Radford et al., 2021), and SigLIP (Zhai et al., 2023)) to capture visual features from the input images or videos. Then, various structures (*e.g.* linear projector (Liu et al., 2024b; Zhu et al., 2023), Q-Former (Li et al., 2023d)) are utilized to project the visual features into the language domain. When it comes to training, BLIP (Li et al., 2022, 2023d) and InstructBLIP (Dai et al., 2023) collect millions data from CC3M (Sharma et al., 2018), SBU (Ordonez et al., 2011), and Conceptual 12M (Changpinyo et al., 2021) for pre-training. LLaVA (Liu et al., 2024b), ALLaVA (Chen et al., 2024), Otter (Li et al., 2023a), ShareGPT4V (Chen et al., 2023), and LLaVA-OV (Li et al., 2024) propose various data engines to generate generic instruct-tuning data for fine-tuning. Furthermore, some techniques have also been developed to boost various downstream abilities of VLLMs, for example, anti-hallucination (Li et al., 2023e; Hu et al., 2023), OCR (Ye et al., 2023a), and prompt-processing (Zhao et al., 2024). Despite their successes, these models mainly focus on the generic ability and are ineffective in processing rare video content and discovery policies.

Table 2: A qualitative comparison between VENUS and major existing methods from three dimensions: data engine, VCD and generic VQA capabilities.

Method	Data engine	VCD capability	Generic VQA capabilities
CLIP	Vision-language pairs	Depend on the seeds	✗
LLaVA	Generic QA	Low	Medium
LLaVA-OV	Knowledge pre-training	Medium	High
LLaVA-Guard	Human-labeled 5K	Discovery tasks	✗
VENUS (Ours)	APSA	High	High

A.3 Vision Language Large Model Evaluation Datasets

Nowadays, many evaluation datasets have been constructed to evaluate VLLM performance. These datasets can be roughly divided into two categories: generic and specific. The generic datasets try to comprehensively review VLLM's various capabilities, for example, MME (Fu et al., 2023), VideoMME (Fu et al., 2024), MMBench (Liu et al., 2023b) etc. In contrast, specific ones aim at one major aspect, like anti-hallucination (Li et al., 2023e; Hu et al., 2023), OCR (Singh et al., 2019), reasoning (Yu et al., 2023) and so on. Obviously, these datasets of evaluating common abilities in real life cannot cover some rare and uncommon Web cases, and are irrelevant to VCD. Therefore, constructing new datasets for the VCD task is urgently necessary.

A.4 Comparison Between Our Work and Existing Approaches

To clarify the differences between VENUS and typical existing methods, in Tab. 2 we present a qualitative comparison from three dimensions: the data engine used to generate training data and models' VCD and generic VQA capabilities. From Tab. 2 we can see that CLIP is designed for similarity match and cannot do VQA. When it comes to the VCD capability, the model performance is influenced by the quality of the seeds. As for the recent off-the-shelf VLLM methods LLaVA and LLaVA-OV, although LLaVA-OV succeeds in lifting VCD and generic VQA capabilities with the help of new proposed knowledge pre-training stage, its VCD performance is limited because of the rare-to-see VCD policies and contents. In contrast, Our VENUS has a new data engine called automatic policy-guided sequential annotator (ASAP). ASAP not only offers rich reasoning texts for different VCD policies but also evaluates the generated data. Extensive experiments show the advantages of VENUS over the existing approaches.

B Ablation Study

In this section, we conduct extensive ablation studies to demonstrate the advantages of the VLLM design of VENUS. Because of the cost of running large-scale VCD evaluation and the correlation between VCD and general performance, we report the results on 4 common general evaluation datasets — TextVQA, MME-P, SEED-Img, and MM-Vet. All the results are in Tab. 3.

B.1 The effect of visual feature extractor

We start by checking the selection of the visual feature extractor (VFE). In this paper, we consider two widely used extractors, which are CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023). As shown in the 1st and the 2nd rows in Tab. 3, SigLIP wins 3 tasks among all the 4 tasks compared with CLIP. Ergo, we choose SigLIP as the visual feature extractor in VENUS.

B.2 The effect of various LLMs

Then, we explore the LLM selection. Here, we consider a total of 4 different LLMs, which are Vicuna-1.5 (Zheng et al., 2024), Mistral-0.2 (Jiang et al., 2023), QWEN2 (Yang et al., 2024) and its upgraded version QWEN2.5. Experimental results correspond to the 1st, 3rd, 4th, and 5th rows of Tab. 3. Obviously, all the best results are achieved by QWEN2.5. Therefore, we install QWEN2.5 in VENUS.

B.3 The effect of dynamic high-resolution technique

Here, we check the effect of the dynamic high-resolution (DHC) technique used to enhance visual resolution. To this end, we compare two variants with/without DHC. As shown in the 2nd and 6th rows in Tab. 3, DHC significantly boosts the performance from 61.8/1524.3/71.6/37.0 to 66.6/1625.5/73.6/36.3. This shows the advantage of DHC.

B.4 The effect of multi-stage training

Subsequently, we install the multi-stage training (MST) to the baseline model, corresponding to the 7th row of Tab. 3. Comparing the results in the 6th and 7th rows, MST lifts the model performance on TextVQA (from 66.6 to 70.4) and MM-Vet (an noticeable improvement of 18.6). This indicates that MST is beneficial for VENUS.

B.5 The effect of batch size

Here we check the effect of the batch size selected in VENUS. Two typical settings adopted by LLaVA (Liu et al., 2024b) and LLaVA-OV (Li et al., 2024) are 128 and 256, respectively. In this paper, we consider four different batch sizes, which are 80, 128, 160, and 256. As shown in the 2nd, 8th, 9th, and 10th rows of Tab. 3, we can see that when picking batch size=160, the win rate against 80, 128, and 256 are all 2:2. This justifies that 160 is a good hyper-parameter.

B.6 The effect of APSA

Here, we step into the ablation of the proposed automatic policy-guided sequential annotator (APSA). Tab. 4 presents the experimental results.

From the 1st and 2nd rows in Tab. 4 we can see that 1) APSA does not impair general performance, and 2) APSA can significantly boost VCD performance. For example, the AP/R@P0.5 values are significantly lifted from 0.379/0.369 to 0.746/0.788. These results indicate that APSA has the ability to efficiently enhance video content discovery without compromising the general capabilities. Furthermore, scaling up training data with APSA can also lift VCD performance. For example, as shown in the 2nd and 3rd rows in Tab. 4, 0.7M version outperforms 6K version in VCD performance.

B.6.1 The effect of quality evaluation in APSA

As described in Sec. 2, we design a quality evaluation phase in ASAP to ensure the high quality of the generated data. To evaluate the performance of the quality evaluation phase, we generate 12K data. For all these 12K data, this quality evaluation phase rejects half of them and accepts 6K data for training. Here, we remove the quality evaluation

Table 3: Ablation studies on the design of visual feature extractor (VFE), LLM, multi-stage training (MST), dynamic high-resolution (DHC), and batch size (BS).

ID	VFE	LLM	MST	DHC	BS	TextVQA	MME-P	SEED-Img	MM-Vet
1	CLIP	QWEN2	✗	✗	128	58.6	1562.2	69.7	35.6
2	SigLIP	QWEN2	✗	✗	128	61.8	1524.3	71.6	37.0
3	CLIP	QWEN2.5	✗	✗	128	62.6	1549.7	71.2	53.8
4	CLIP	Vicuna1.5	✗	✗	128	58.2	1510.7	66.1	31.1
5	CLIP	Mistral0.2	✗	✗	128	54.8	1403.5	67.0	30.1
6	SigLIP	QWEN2	✗	✓	128	66.6	1626.5	73.6	36.3
7	SigLIP	QWEN2	✓	✓	128	70.4	1624.3	74.0	54.9
8	SigLIP	QWEN2	✗	✗	80	61.8	1524.3	71.6	37.0
9	SigLIP	QWEN2	✗	✗	160	61.1	1590.5	71.8	35.8
10	SigLIP	QWEN2	✗	✗	256	61.5	1592.5	71.2	35.6
11	SigLIP	QWEN2.5	✓	✓	160	72.4	1590.7	74.9	56.5

Table 4: Ablation study on the data engine utilized for data generation. We report the average AP and R@P0.5 of all 13 policies for VCD.

Configuration	TextVQA	MME-P	SEED-Img	MM-Vet	AP	R@P0.5
Baseline	72.8	1608.5	74.9	56.8	0.379	0.369
+0.7M qualified APSA data	73.2	1618.8	75.2	54.8	0.565	0.573
+6K qualified APSA data	72.4	1590.7	74.9	56.5	0.482	0.491
+12K APSA data without qualification	71.6	1580.0	75.0	50.7	0.464	0.483

phase and utilize all the 12K data as a variant. The corresponding results are given in the 4th row of Tab. 4. Obviously, both the general performance and the VCD performance are inferior to the 6K version. For example, MM-Vet and AP of VCD are deteriorated from 56.6/0.482 to 50.7/0.464, respectively. This indicates that increasing the amount of data without considering quality cannot guarantee desirable performance.

C Results on more benchmarks

Here, we give results on more evaluation benchmarks to better demonstrate the generality of our VENUS system. Accordingly, in addition to the 13 discovery policies in our VCD-Bench, we test a total of 10 general VLLM benchmarks, including VQAv2 (Goyal et al., 2017), GQA (Hudson and Manning, 2019), TextVQA (Singh et al., 2019), POPE (Li et al., 2023e), MME (Fu et al., 2023), MMBench (Liu et al., 2023b), SEED-Bench (Li et al., 2023b), MM-Vet (Yu et al., 2023), Original CIEM (CIEM-Org) (Hu et al., 2023), and Prompt-augmented CIEM (CIEM-Aug) (Zhao et al., 2024). All the experimental results are given in Tab. 5.

From Tab. 5 we can see that VENUS is superior to LLaVA-OV in not only VCD tasks but also a wide scope of VQA tasks like GQA, MME, MM-Vet, CIEM-Org, and CIEM-Aug. As a result, VENUS wins 7 different tasks. This indicates the strong performance of our VENUS system.

D Case Study of APSA

To better understand the advantages of our proposed APSA, we provide three cases (two for different kinds of animals and one for gambling policy) comparisons with the typical data engine that directly requires the VLLM to answer the question. As shown in the 1st case in Fig. 6, when querying Larvivora cyane, a rare bird, due to lack of specific knowledge, the VLLM may mistakenly recognize it as a more regular bird – blue jay. In contrast, with the help of our ASAP, the VLLM can provide detailed and correct reasoning to recognize Larvivora cyane. Similar results can also be observed from the rest two cases.

E Visual Case of VENUS

In this section, we provide a visualization of VENUS on the aforementioned image quality discerning policy to better demonstrate the workflow of our system. As shown in Fig. 7, given a detail text policy that describes the definition and expecta-

Table 5: Performance comparison between VENUS and state-of-the-arts on various discovery and generic benchmarks. The metrics for VCD are AP/R@P0.5. '-' means the corresponding method cannot do this task.

Datasets		VENUS	LLaVA-OV	LLaVA-NeXT	ShareGPT4V	LLaVA-1.5
VCD	AP	**0.565**	0.430	0.202	0.309	0.360
	R@P0.5	**0.573**	0.475	0.153	0.301	0.315
VQA	VQAv2	84.0	**84.5**	82.2	80.6	81.1
	GQA	**63.3**	62.2	64.8	63.3	63.4
	TextVQA	73.2	**73.8**	65.7	60.4	62.6
	POPE-All	87.8	**89.1**	86.7	86.8	87.5
	MME-P	**1618.8**	1584.8	1502.9	1567.4	1549.7
	MMBench-En	79.9	**80.8**	68.7	68.8	76.1
	SEED-Img	75.2	**75.4**	72.2	69.7	71.2
	MM-Vet	**54.8**	52.7	47.3	37.6	53.8
	CIEM-Org	**90.6**	90.5	88.5	89.3	89.9
	CIEM-Aug	**87.8**	87.6	84.0	82.7	87.0
#Total Wins		**7**	5	0	0	1

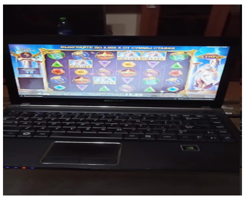
Case 1

By typical data engine:
User: Does this photo contains any Larvivora cyane?
Agent: No, there is no Larvivora cyane. The bird in the image is a Blue Jay.
By ours:
The bird in the image is indeed a Larvivora cyane, commonly known as the Siberian Blue Robin. The key characteristics that confirm this identification include: Coloration: … Size and Shape: ... Distinctive Features: ….

Case 2

By typical data engine:
User: Does this photo contains any eels?
Agent: Yes, the photo contains an eel. The fish in the image has a long, slender body and lacks pelvic fins, which are characteristic features of eels.
By ours:
The image likely does not contain an eel because the creature shown exhibits characteristics inconsistent with eels ….

Case 3



By typical data engine:
User: Does this photo contain any gambling content?
Agent: No, the image shows a laptop screen displaying a video slot machine game.

By ours:
Yes, this image contains gambling content. The screen displays a slot machine …

Figure 6: Visualization comparisons between a typical existing data engine and our APSA.

tions of low-quality, VENUS can correctly analyze the visual inputs and provide scores and textual reasons. Thus, the scores can be storaged as a feature for downstream applications like retrieval and recommendation.

Policy:
You are an expert in discerning low-quality images. According to business requirements, there are 3 categories of low-quality image classification: Border Image, Text Image, and Stitched Image.
The definitions of each category are as follows:
1. The definition of a Border Image refers to images with the following characteristics:
a. The image has obvious border lines or frames at the top, bottom, left, or right;
b. The border can clearly distinguish the content of the image from the background, enhancing visual focus;
c. The top, bottom, left, or right of the image has a solid color area or a frosted glass area;
2. The definition of a Text Image is:
a. There is text on the image and the text does not belong to the photo shooting scene itself, it is made by additional post-production;
Note: If the text is included in the subject of the shot itself, such as the brand name on a cup, or the shop name captured in the shot, it is not a Text Image;
b. There is a watermark on the image;
If an image meets any of the above two definitions of a Text Image, it can be considered a Text Image;
3. The definition of a Stitched Image is: The image is made up of 2 or more stitched images;
If it meets any of the low-quality categories, it is a low-quality image.
Input an image, please determine whether it is a low-quality image.

Text output: The image contains text in the bottom right corner that appears to be a watermark. This text does not belong to the photo shooting scene itself and is made by additional post-production. Therefore, this image meets the definition of a Text Image. Therefore, the answer is yes, this is a low-quality image.
Yes score: 0.834

Text output: The image does not have any obvious border lines or frames at the top, bottom, left, or right. There is no solid color area or frosted glass area that distinguishes the content from the background. Additionally, there is no text or watermark present on the image, and it is not made up of stitched images. Therefore, the answer is no, this is not a low-quality image.
Yes score: 0.002

Figure 7: Visualization of VENUS on low-quality image discerning policy.