Leveraging Vision Language Models for Understanding and Detecting Violence in Videos

Jose Alejandro Avellaneda Gonzalez[a], Tetsu Matsukawa[b] and Einoshin Suzuki[c]
ISEE, Kyushu University, Fukuoka, 819-0395, Japan
joseavellaneda88@gmail.com, {matsukawa, suzuki}@inf.kyushu-u.ac.jp

Keywords: Video Violence Detection, Vision-Language Models, Large Language Models (LLMs), Video Violence Analysis.

Abstract: Detecting violent behaviors in video content is crucial for public safety and security. Ensuring accurate identification of such behaviors can prevent harm and enhance surveillance. Traditional methods rely on manual feature extraction and classical machine learning algorithms, which lack robustness and adaptability in diverse real-world scenarios. These methods struggle with environmental variability and often fail to generalize across contexts. Due to the nature of violence content, ethical and legal challenges in dataset collection result in a scarcity of data. This limitation impacts modern deep learning approaches, which, despite their effectiveness, often produce models that struggle to generalize well across diverse contexts. To address these challenges, we propose VIVID: Vision-Language Integration for Violence Identification and Detection. VIVID leverages Vision Language Models (VLMs) and a database of violence definitions to mitigate biases in Large Language Models (LLMs) and operates effectively with limited video data. VIVID functions through two steps: keyframe selection based on optical flow to capture high-motion frames, and violence detection using VLMs to translate visual representations into tokens, enabling LLMs to comprehend video content. By incorporating an external database with definitions of violence, VIVID ensures accurate and contextually relevant understanding, addressing inherent biases in LLMs. Experimental results on five datasets—Movies, Surveillance Fight, RWF-2000, Hockey, and XD-Violence— demonstrate that VIVID outperforms LLM-based methods and achieves competitive performance compared with deep learning-based methods, with the added benefit of providing explanations for its detections.

1 INTRODUCTION

Detecting violent behaviors in video content is essential for public safety and security research. Video violence detection aims to identify actions involving physical force that harm individuals or damage property, such as fighting and assault (Ullah et al., 2023). It has various applications in video surveillance, such as automated systems that can promptly deliver timely alerts for hazardous scenarios, enabling swift responses (Mumtaz et al., 2023).

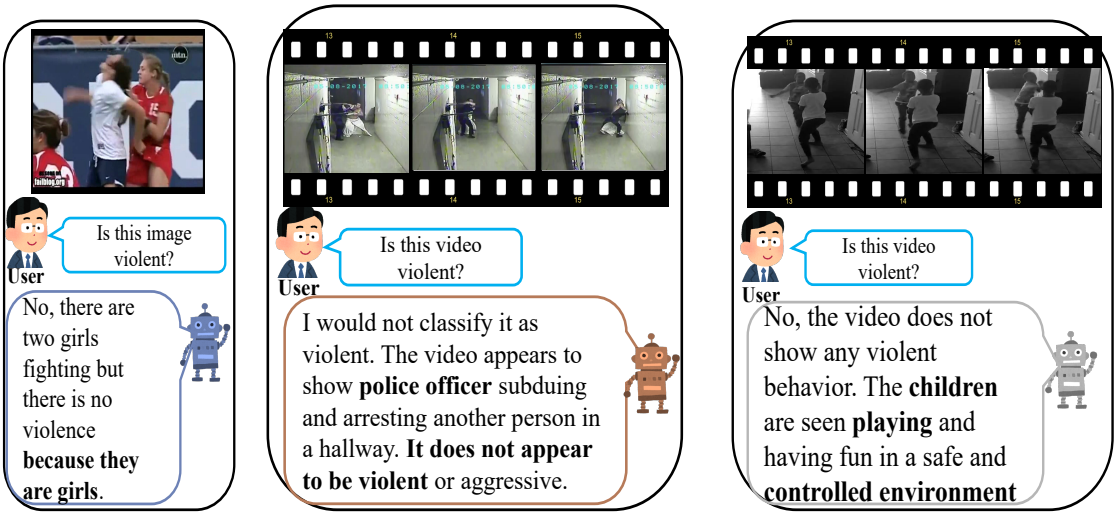
Traditional methods primarily focus on manual feature extraction from video data (De Souza et al., 2010; Bermejo et al., 2011; Deniz et al., 2014; Senst et al., 2017; Das et al., 2019; Febin et al., 2020). These methods manually extract spatial, temporal, or spatiotemporal features and use supervised or unsupervised classification methods to determine whether the content is violent. However, these methods exhibit deficiencies in robustness and adaptability in real-world scenarios due to limitations such as installation angles, diverse locations, varying backgrounds, and video resolutions (Park et al., 2024).

Several works (Park et al., 2024; Abdali and Al-Tuma, 2019; Sudhakaran and Lanz, 2017; Cheng et al., 2021; Aktı et al., 2019) explore deep learning models that autonomously identify features and patterns to classify violent behavior. However, datasets specifically related to video violence detection are scarce (Mumtaz et al., 2023). Despite their favorable performance, the practical applicability of these methods heavily relies on the training data. Training deep models or learning motion patterns from an insufficient set of violent videos results in non-generic models which may not be practical for real-world scenarios. Therefore, current violent detection methods are insufficient for generalizing and identifying vio-

[a] https://orcid.org/0009-0008-3082-8921
[b] https://orcid.org/0000-0002-8841-6304
[c] https://orcid.org/0000-0001-7743-6177

Gonzalez, J. A. A., Matsukawa, T. and Suzuki, E.
Leveraging Vision Language Models for Understanding and Detecting Violence in Videos.
DOI: 10.5220/0013160000003912

In *Proceedings of the 20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2025) - Volume 2: VISAPP*, pages 99-113
ISBN: 978-989-758-728-3; ISSN: 2184-4321



(a) BLIP-2. (b) Video-LLaMA. (c) Video-LLaVA.

Figure 1: LLM's inherited bias examples: they contain gender, occupation, age, race, and other biases.

lence in real-world contexts.

Recently, several efforts have been made to construct frameworks for visual understanding based on Vision Language Models (VLMs) (Li et al., 2023a; Dai et al., 2023; Panagopoulou et al., 2023; Lin et al., 2023; Li et al., 2023b). These models take images or videos as input and utilize Large Language Models (LLMs) to provide descriptions or answer questions based on the visual content.

However, LLMs inherit biases from their training data, impacting their ability to generalize across various problems, including violence detection. For instance, as shown in Figure 1b, consider a scenario where the input of the VLM is a video of a police officer subduing and arresting someone. This scenario contains elements of violence, such as physical force and restraint, which can cause harm or distress. The VLM classified this scenario as non-violent due to an inherent bias from the training data, reflecting societal perceptions of law enforcement actions. This misclassification arises from the LLM's reliance on biased data, skewing its interpretation of violent behavior.

Furthermore, guardrails in LLMs, aligned with ethical standards, may lead to ambiguous answers on sensitive topics. Fine-tuning video understanding methods to align with video content (Lin et al., 2023; Li et al., 2023b) can further alter their understanding. Addressing bias and ensuring robustness across different scenarios remains an ongoing challenge.

To overcome these challenges, we propose a novel method called VIVID: **V**ision-Language **I**ntegration for **V**iolence **I**dentification and **D**etection. VIVID is designed for scenarios where video or image data is scarce or insufficient. Specifically, our approach leverages a VLM in conjunction with an external knowledge violence database. This combination helps address inherent biases and improves the detection and identification of violent behaviors by providing accurate and contextually relevant information.

VIVID involves two main steps: key-frame selection and violence detection. In the first step, we analyze the optical flow within the video to capture essential moments that may potentially contain forceful actions capable of causing harm or damage.

In the second step, we leverage a VLM and an LLM to serve as our violence detector. VLM models are pre-trained on large datasets containing paired images and texts, allowing them to understand and generate texts based on images. LLMs, on the other hand, are designed to comprehend and generate human-like language, meaning that if a concept such as violence can be semantically defined, an LLM would have the ability to identify the concept in a given context.

We propose to address the intrinsic biases of LLMs by using an external knowledge base that contains definitions of violence and violence-related terms. We handle the bias in two main steps: (1) identifying the most relevant violence-related definition associated with the video content, and (2) combining this definition with a video representation to enhance the LLM's accuracy related to violence. These steps ensure that the LLM's responses are both accurate and contextually relevant. VIVID requires no additional training and small computational overhead during inference, making it efficient and practical for real-world applications. Additionally, VIVID not only classifies violent content but also provides an explanation in text, allowing users to understand why the content is deemed violent.

In summary, our contributions are as follows:

- We propose VIVID, a novel method that integrates Vision-Language Models (VLMs) with Language Models (LLMs) for robust and interpretable video violence detection.
- We offer a zero-shot classification alternative, allowing the model to classify violent content without additional training, making it particularly suitable for scenarios with limited labeled data.
- We propose a multimodal retrieval method to compare visual and text features, addressing biases inherent in LLMs and leading to more accurate and contextually relevant results.
- By combining VLMs with an external knowledge base, our method effectively captures complex visual and contextual cues, improving its accuracy in detecting violence.
- Our framework provides clear explanations for its classifications, enhancing user trust and understanding its decisions.
- We demonstrate that the proposed method consistently outperforms other methods based on LLMs across multiple datasets, highlighting its effectiveness in identifying violent content.

2 RELATED WORK

2.1 Violence Detection

Violence detection in videos is a critical area of research with significant implications for public safety and surveillance. Traditional approaches to violence detection have predominantly relied on manually crafted features and classical machine learning algorithms. Early methods like those proposed by De Souza et al. (De Souza et al., 2010) and Deniz et al. (Deniz et al., 2014) focused on extracting spatio-temporal features from video frames to identify violent actions. For instance, De Souza et al. (2010) (De Souza et al., 2010) utilized local spatio-temporal features with the Bag of Visual Words (BoVW) representation, extracting features using descriptors such as Scale-Invariant Feature Transform (SIFT). These features were then classified using Support Vector Machines (SVMs) to identify violent actions. Similarly, Deniz et al. (2014) (Deniz et al., 2014) developed a method for fast violence detection by extracting extreme acceleration patterns through the application of the Radom transform to the power spectrum of consecutive frames, using SVMs to rapidly identify violent events.

Senst et al. (2017) (Senst et al., 2017) employed global motion-compensated Lagrangian features and scale-sensitive video-level representation to capture motion patterns and dynamics within video footage. This method utilized histogram-intersection-based clustering to detect instances of violence effectively. Das et al. (2019) (Das et al., 2019) utilized Histogram of Oriented Gradient (HOG) features to capture edge and gradient information from video frames. These features were used with classifiers such as SVM, Logistic Regression, Random Forest, Linear Discriminant Analysis (LDA), Naïve Bayes, and K-Nearest Neighbors (KNN) are used for classification purposes.

Hassner et al. (2012) (Hassner et al., 2012) introduced the concept of violent flows, utilizing optical flow and motion regions to detect violent activities in real time. This approach involved analyzing motion patterns within video frames to identify sudden and aggressive movements, which were then classified using SVMs.

However, these traditional methods exhibit several limitations when applied to real-world scenarios. They often struggle with robustness and adaptability to diverse environmental factors, such as different installation angles, backgrounds, and video resolutions. For instance, a video captured from a high angle might obscure crucial details, while a cluttered environment may introduce noise, complicating feature extraction and analysis. Consequently, there has been a shift towards deep learning-based methods in recent years to address these challenges and improve the accuracy and reliability of violence detection systems.

Recent advancements in deep learning have led to significant improvements in violence detection systems. These methods leverage the power of neural networks to automatically learn and extract features from video data, often leading to superior performance compared to traditional approaches.

One of the latest methods is by Park et al. (2024) (Park et al., 2024), who proposed a Conv3D-based network that combines optical flow and RGB data to detect violent behaviors in videos. This method utilizes three-dimensional convolutions to capture spatio-temporal features from video frames, providing a comprehensive understanding of motion and appearance. Similarly, Abdali and Al-Tuma (2019) (Abdali and Al-Tuma, 2019) introduced a robust real-time violence detection system using Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks. By capturing both spatial and temporal features, this method effectively enables the detection of violence in video sequences.

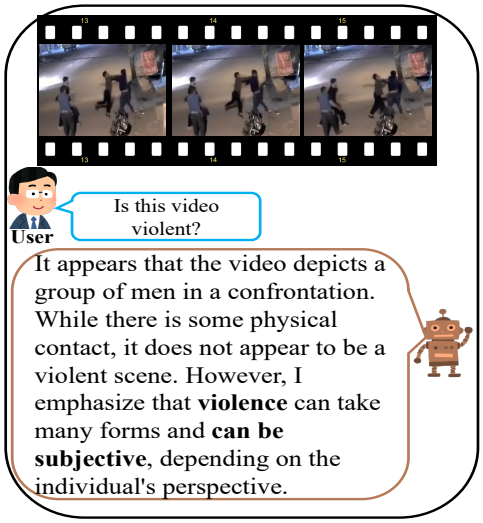


Figure 2: LLM's guardrail bias example. This example shows an ambiguous answer from Video-LLaMA.

Sudhakaran and Lanz (2017) (Sudhakaran and Lanz, 2017) developed a method leveraging convolutional LSTM networks to detect violent activities. The convolutional layers extract spatial features, which are then processed by LSTM units to capture temporal patterns. Furthermore, Wu et al. (2020) (Wu et al., 2020) proposed a multimodal approach that combines audio and visual information to detect violence under weak supervision, particularly in scenarios where one modality might be less effective

Traoré et al. (2020) (Traoré and Akhloufi, 2020) used a combination of Deep Recurrent Neural Networks (RNNs) and CNNs to detect violence in videos. The CNNs are responsible for extracting spatial features from video frames, while the RNNs, particularly LSTM networks, capture the temporal dynamics.

Another approach by Ullah et al. (2019) (Ullah et al., 2019) leverages 3D Convolutional Neural Networks (3D CNNs) to extract spatiotemporal features from video sequences for violence detection. The 3D CNN captures both spatial and temporal dimensions of violent actions.

However, these methods are often limited by the scarcity of violence-specific datasets, resulting in non-generalizable models. Training these models on a limited amount of data can affect their ability to generalize effectively across various real-world scenarios. Additionally, the sensitive nature of violent content poses challenges for dataset collection due to ethical and legal considerations, further complicating the development of robust and generalizable models.

2.2 Vision Language Models

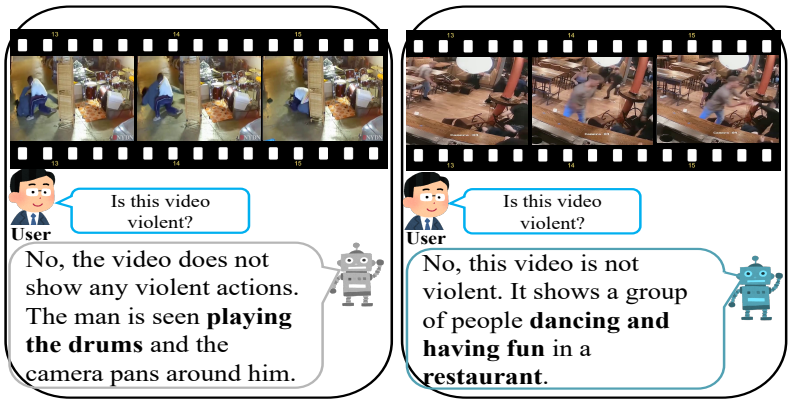
Vision Language Models for image understanding (Li et al., 2023a; Dai et al., 2023) and video understanding (Zhang et al., 2023; Lin et al., 2023; Li et al., 2023b; Panagopoulou et al., 2023) process images or videos as input and leverage LLMs to generate descriptions or respond to queries based on the visual content. This integration of visual and textual data facilitates a more holistic comprehension of the content.

The Salesforce Research Division proposed the "BLIP family models," which include BLIP-2 (Li et al., 2023a), InstructBLIP (Dai et al., 2023), and X-InstructBLIP (Panagopoulou et al., 2023). These models share a common goal of integrating vision and language models and building upon each other's architectures and concepts. For instance, BLIP-2 (Li et al., 2023a) is a vision-language pre-training method that bootstraps from frozen pre-trained unimodal models to bridge the modality gap by using a Querying Transformer (Q-Former). The Q-Former is a trainable module designed to connect a frozen image encoder with a frozen LLM. Similarly, InstructBLIP (Dai et al., 2023) extends BLIP-2 with the aim of creating general-purpose vision-language models through instruction tuning, thereby enhancing the model's ability to follow natural language instructions. Furthermore, X-InstructBLIP (Panagopoulou et al., 2023) extends the capabilities of InstructBLIP by enabling cross-modal reasoning across various modalities, including video, without extensive modality-specific pre-training.

Similarly, Video-LLaMA (Zhang et al., 2023) enables frozen LLMs to understand both visual and auditory content in videos. It features two branches: the Vision-Language Branch and the Audio-Language Branch, which convert video frames and audio signals into query representations compatible with LLMs enhancing its ability to comprehend complex video content.

However, these methods rely entirely on frozen LLMs, which introduces potential biases inherited from the training data. These biases can significantly affect the model's capacity to generalize across diverse contexts, including the detection of violent content. For example, suppose the training data encompasses biases related to specific demographics or scenarios. In that case, the model may encounter difficulties in accurately interpreting or responding to novel, unseen data that diverges from these patterns as shown in Figure 1. This issue is particularly critical in applications such as violence detection, where an accurate and unbiased interpretation is essential.

Moreover, many LLMs are equipped with



(a) Video-LLaVA. (b) LLaMA-VID.

Figure 3: Example of LLM's alignment bias.

guardrails intended to align with ethical standards and societal expectations. These guardrails are designed to prevent generation of harmful or inappropriate content. However, these measures can inadvertently introduce additional biases (Dong et al., 2024). For example, when dealing with sensitive topics such as violence, LLMs may generate ambiguous responses to avoid potential controversy. This ambiguity can hinder the model's effectiveness in providing clear and precise answers, particularly in critical applications like violence detection or other safety-related tasks, as illustrated in Figure 2.

Other methodologies, such as those proposed in Video-LLaVA (Lin et al., 2023) and LLaMA-VID (Li et al., 2023b), involve not only training a model to bridge the gap between visual and language modalities but also fine-tuning the LLMs to align with video content for better understanding. For instance, Video-LLaVA (Lin et al., 2023) is a model that handles both images and videos. It aligns image and video representations into a unified visual feature space, enabling an LLM to learn from this unified visual representation. Additionally, LLaMA-VID (Li et al., 2023b) is a method where each frame is transformed into two distinct tokens: the context token, which captures the overall high-level context, and the content token, which focuses on specific visual details. During training, the LLM learns to associate these tokens with the corresponding visual and textual data.

These adaptations entail modifying the model's parameters based on the specific attributes of the video data. While this process can enhance the model's performance in comprehending and interpreting video content, it can also alter the LLM's understanding within specific contexts. Consequently, the model's responses may become excessively tailored to the content observed in the videos used for fine-tuning, potentially diminishing its ability to generalize to new, diverse scenarios as shown in Figure 3.

3 PROBLEM FORMULATION

Our objective is to detect violence within video clips in the context of human monitoring. Given the scarcity, diversity, and limited availability of labeled data specifically containing violent content, we approach this problem as a zero-shot violence detection scenario, i.e., without prior training on video clips from target datasets that include violence or non-violence classes.

Following this paradigm, the model receives as input a set of video clips $\{(\mathbf{C}_i)\}_{i=1}^{n}$. For $\mathbf{C}_i$, the model produces two outputs: an associated class prediction $\hat{y}_i \in \{0,1\}$ and an explanatory text $\mathbf{e}_i$ detailing the classification decision. In this context, a class label 0 represents non-violent content, while 1 denotes violent content.

Our focus lies in the specific detection of physical violence defined as an act attempting to cause, or resulting in, pain and/or physical injury, or damage to the state of something (American Psychological Association, 2024).

To assess the effectiveness of our method, we employ accuracy and F1-score. These measures collectively provide insights into the model's performance, ensuring robustness and reliability in detecting violent content within video clips.

4 PROPOSED METHOD

4.1 Overview

We propose VIVID: **V**ision-Language **I**ntegration for **V**iolence **I**dentification and **D**etection. This method

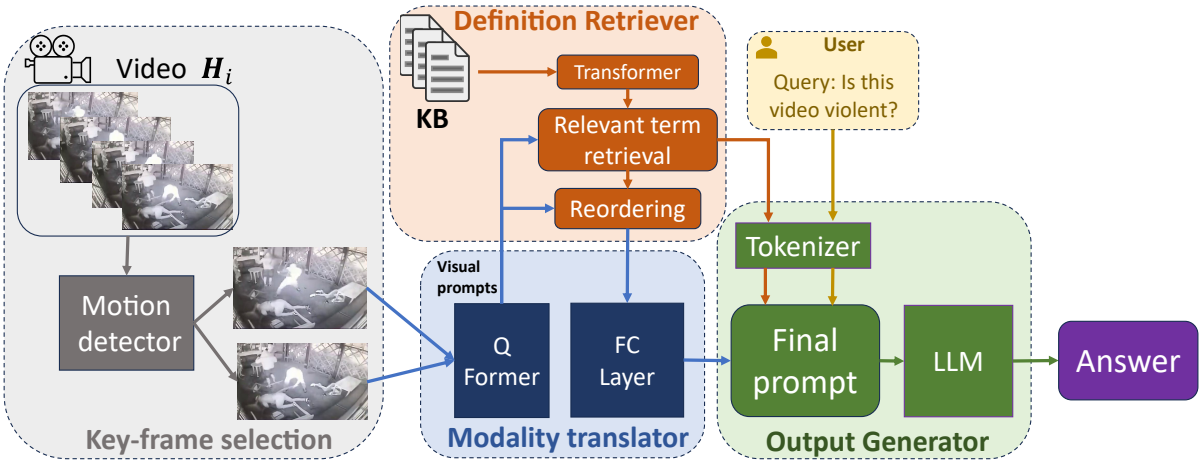


Figure 4: Architecture of the proposed method.

Input: Video clip $\mathbf{C}$, Knowledge base **KB**, Number of frames k
Output: Prediction $\hat{y}$ and Explanation $\mathbf{e}$

// Key-frame Selection;
key_frames $\leftarrow$ Top k frames with the greatest magnitude of motion;

// Rest: Violence Detector;
// Modality Translator;
queries $\leftarrow$ [];
for *each frame* $\boldsymbol{I}_t \in$ ***key_frames*** **do**
  $\mathbf{V}_t \leftarrow$ Q_Former($\mathbf{I}_t$);
  append (**queries**, $\mathbf{V}_t$);
end

// Definition Retriever;
definitions $\leftarrow$ [];
for *each query* $\boldsymbol{V}_t \in$ ***queries*** **do**
  $\mathbf{D}_t \leftarrow$ retrieve_definition($\mathbf{V}_t$, **KB**);
  append (**definitions**, $\mathbf{D}_t$);
end
$\mathbf{D}_s \leftarrow$ select_definition(**definitions**);
$\mathbf{V}'_t \leftarrow$ reorder_queries(**queries**, $\mathbf{D}_s$);

// Modality Translator;
$\mathbf{T}_v \leftarrow$ FC_layer($\mathbf{V}'_t$);

// Output Generator;
$\mathbf{Q}_u \leftarrow$ "Is this video violent?";
$\mathbf{T}_u \leftarrow$ LLM_tokenizer($\mathbf{Q}_u$);
$\mathbf{T}_d \leftarrow$ LLM_tokenizer($\mathbf{D}_s$);
final_prompt $\leftarrow$ create_prompt($\mathbf{T}_u, \mathbf{T}_v, \mathbf{T}_d$);

$(\hat{y}, \mathbf{e}_t) \leftarrow$ LLM_process(**final_prompt**);
return $(\hat{y}, \mathbf{e})$

Algorithm 1: Violence Detection using VIVID for a single video clip $\mathbf{C}$.

leverages the strengths of Vision-Language Models (VLMs) and Language Models (LLMs) to provide a robust, interpretable solution for video violence detection.

The detailed steps of the VIVID algorithm, outlined in Algorithm 1. It involve two main processes: key-frame selection and violence detection.

In the first step, key-frame selection, the model utilizes optical flow (Bobick and Davis, 2001) analysis to quantify motion within the video. For each frame $\mathbf{I}_t$ in the video $\mathbf{C}$, the magnitude of motion is computed and stored. The frames with the highest motion activity are selected as keyframes, capturing essential moments that potentially contain violent actions.

In the second step, violence detection, the model performs several sub-tasks to analyze the keyframes. It begins by translating each frame into visual queries $\mathbf{V}_t$ through the Q-Former, enabling direct comparison with text embeddings. The definitions related to each query $\mathbf{D}_t$ are then retrieved from the knowledge base **KB**. The most relevant definition for the video $\mathbf{D}_s$ is then selected from the list that contains all the definitions of each frame. The **append**(x, y) function is used to add the element y to the list x. In this context, it is utilized to add new items to lists **queries** and **definitions** during the processing steps.

$\mathbf{D}_s$ is used to reorder $\mathbf{V}_t$ to mitigate the 'lost in the middle' effect, which will be explained in detail in Section 4.3.2. The reordered queries $\mathbf{V}'_t$ are subsequently processed through a fully connected layer to create visual tokens. A prompt is created using the visual tokens, definition tokens, and user query tokens. The final prompt is then processed by the language model to produce a label indicating whether the content is violent or not, along with an explanatory text detailing the classification decision.

A summary of the architecture is shown in Figure 4.

4.2 Key-Frame Selection

As we argued in Section 1, violent content is scarce and has limited availability. This scarcity makes it challenging to train models that can generalize patterns associated with violent actions. To address this issue, we propose using optical flow to extract potential violent frames.

Optical flow (Bobick and Davis, 2001) effectively captures dynamic movements in video sequences, which are often indicative of violent behaviors. By semantic definition, a violent action involves forceful behaviors like hitting, kicking, or grappling, which inherently exhibit a greater degree of motion compared with other non-violent activities (Herrenkohl, 2011). Consequently, we extract potential violent frames by calculating the norm of dense optical flow vectors, summarizing the overall motion magnitude.

Several alternatives to optical flow could be considered for frame extraction, including traditional approaches like Frame Differencing and Background Subtraction (Szeliski, 2022), as well as more advanced techniques such as Motion History Images (MHI) (Bobick and Davis, 2001). For instance, Frame Differencing calculates the difference between consecutive frames to detect changes in the scene. While simpler and computationally less expensive, it may miss subtle movements and fail to capture complex motion patterns associated with violent actions (Szeliski, 2022). Similarly, Background Subtraction isolates moving objects by subtracting a background model from each frame. However, it is susceptible to lighting changes and background clutter, leading to false detections (Szeliski, 2022). MHIs create a motion that represents a single image over time by accumulating motion information. However, they may blur detailed motion information and do not provide specific frames in the video (Sun et al., 2022), making it challenging to identify and extract frames corresponding to specific violent actions.

Unlike these alternatives, optical flow offers a balanced approach that effectively captures dynamic motion. It quantifies the velocity and direction of motion at each pixel, providing detailed information about movements within the scene (Szeliski, 2022). This granularity is essential for identifying violent actions, which typically involve rapid and forceful movements. Additionally, optical flow is computationally efficient and can be applied to both low-resolution and high-resolution video data.

The importance of using optical flow in violence detection scenarios has been emphasized in several works (Dalal et al., 2006; Hassner et al., 2012; Wang et al., 2013; Park et al., 2024). These methods utilize optical flow to extract trajectories, build descriptors, or extract features from video data. However, they primarily use optical flow for training models to learn patterns from violent video content, whereas our approach focuses on selecting key-frames directly based on motion magnitude.

Consider a video clip $\mathbf{C}$ that is decomposed into a set of frames, where a frame at time t, represented as $I(x,y,t)$, with (x,y) being the pixel coordinates. The dense optical flow relates the partial derivatives of image intensity with respect to spatial coordinates and time as follows:

$$I_x \cdot u_x + I_y \cdot u_y + I_t = 0, \tag{1}$$

where, I_x and I_y are the spatial gradients of image intensity in the x and y directions, respectively, and I_t is the temporal gradient. u_x and u_y represent the flow velocities in the x and y directions, respectively. The norm of optical flow for a single pixel is calculated as:

$$|u| = \sqrt{(u_x)^2 + (u_y)^2}. \tag{2}$$

By summing the norms across all pixels in the frame, we obtain the total motion magnitude for the frame as:

$$U_t = \sum_x \sum_y |u(x,y,t)|. \tag{3}$$

After calculating U_t for different frames along the video, we extract the top k frames with the highest values, which we assume correspond to the frames containing the highest activity. These frames serve as input for the subsequent analysis.

We acknowledge that extracting keyframes with the highest activity does not guarantee selecting all violent frames. For instance, high motion, such as a passing car, might deceive the optical flow selector, causing it to neglect frames with slow-motion violence, such as a person threatening another with a knife. In such cases, VIVID may fail to detect the violent scenario. We leave this challenge to our future work, which can be addressed by incorporating the detection of potentially dangerous objects during the frame selection process.

4.3 Violence Detection

In this step, we combine a Vision Language Model (VLM) with a Language Model (LLM) to detect violence in videos. The detection process involves several key modules working together, as illustrated in Figure 4.

The Modality Translator component translates visual data from the video frames into sequence of tokens, which can be processed by the LLM. This involves using the VLM's Q-Former to generate visual queries $\mathbf{V}_t$. After reordering the queries $\mathbf{V'}_t$, they are processed by a fully connected (FC) layer that converts them into tokens $\mathbf{T}_\mathrm{v}$ compatible with the LLM. This translation ensures that the visual information is appropriately represented for subsequent analysis.

Following this, the Retrieval module plays a critical role. It retrieves relevant information by comparing $\mathbf{V}_t$ with the definitions in the knowledge base **KB**, identifying the most pertinent violence-related definitions $\mathbf{D}_t$. From these, we extract the most relevant definition $\mathbf{D}_\mathrm{s}$ for the entire video. To address the 'lost in the middle' effect often observed in LLMs, the visual queries are reordered to prioritize the most relevant information, ensuring the LLM receives a coherent and contextually appropriate set of inputs.

Finally, the Generator module combines $\mathbf{T}_\mathrm{v}$, user query tokens $\mathbf{T}_\mathrm{u}$, and the most relevant definition tokens $\mathbf{T}_\mathrm{d}$ to create a comprehensive final prompt. This prompt is then processed by the LLM, which generates a response to the user's query, determining whether the video contains violent content.

4.3.1 Modality Translator

As shown in Figure 4, we selected an architecture based on BLIP-2 (Li et al., 2023a) for the VLM. This architecture includes a Q-Former, which provides visual queries containing visual representations $\mathbf{V}_t$ corresponding to the text, making them directly comparable with text embeddings in a common subspace. Additionally, the VLM has a fully-connected (FC) layer that linearly projects the reordered visual queries $\mathbf{V}'_t$ to match the input dimension of LLM. This means that the FC layer acts as a visual tokenizer, converting the reordered visual queries into a set of tokens $\mathbf{T}_\mathrm{v}$ that the LLM can process. The reordering of visual queries ensures that the most relevant visual information is emphasized, which is crucial for accurate analysis. This process will be explained in detail in section 4.3.2.

4.3.2 Definition Retriever

In this section, we explain the processes of retrieving the definitions to ensure their relevance and accuracy for violence detection. We break down this section into three sub-modules: Creating KB, where we define and build the knowledge base of violence-related terms and definitions; Relevant Term Retrieval, which covers how visual queries are compared with the knowledge base to extract the most relevant definitions; and Reordering Visual Queries, where we discuss the reordering of visual queries to prioritize the most pertinent information, mitigating the 'lost in the middle' effect in LLMs.

Creating KB. Retrieval Augmented Generation (RAG) (Lewis et al., 2020) enhances LLMs by integrating external knowledge, such as databases, to improve performance in knowledge-intensive or domain-specific applications that require continually updating information (Gao et al., 2023). RAG retrieves relevant documents based on the user's query and combines them with the original prompt to generate a response.

Our proposed method addresses biases within LLMs by drawing inspiration from the RAG framework. However, instead of adding new information, it focuses on mitigating existing biases related to sensitive topics like violence. To achieve this goal, we leverage the VLM output, which provides informative visual representations $\mathbf{V}_t$ corresponding to the text. Instead of relying on user queries to retrieve documents, we compare these visual representations with a predefined set of violence-related terms and definitions, referred to as the knowledge base **KB**.

Since, to the best of our knowledge, there is no existing comprehensive knowledge base of violence-related terms, we propose one that is defined as:

$$\mathbf{KB} = \{(w_1, d_1), (w_2, d_2), \ldots, (w_n, d_n)\}, \quad (4)$$

where w_i represents the i-th violence-related term (word) and d_i represents the corresponding definition.

We created the knowledge base (KB) of violence-related terms, comprising 48 entries, each explaining various aspects of physical violence. These aspects include direct physical contact, violent sports, collective violence, and broader terms denoting physical force.

First, direct physical contact encompasses terms such as "physical abuse", "physical disputes", "rape", "battery", and "fights", describing incidents where physical harm is directly inflicted upon individuals.

Regarding terms that frequently escalate into physical confrontations, this includes "clash", "assault", "aggression", and "confrontation", highlighting situations that may start as verbal or non-physical conflicts but can quickly escalate to physical violence.

In terms of violent sports, these are activities where the objective is to subdue opponents through physical force, such as "boxing", "wrestling", "muay Thai", and "karate", where physical engagement is a structured and accepted part of the sport.

For collective violence, the terms include "riot", "brawl", "vandalism", and "scuffle", reflecting sce-

narios where multiple individuals are involved in violent acts, often leading to public disorder or damage.

Lastly, broader terms denoting physical force encompass words like "war", "savagery", "oppression", "bullying", and "genocide", representing extensive and severe forms of violence, highlighting large-scale or systematic acts of harm and intimidation.

This diverse compilation ensures a comprehensive understanding of violence across different contexts. The definitions were obtained from well-known sources, including the Cambridge and Oxford dictionaries, Wikipedia, and Wex, a legal dictionary sponsored by the Legal Information Institute at Cornell Law School.

Relevant Term Retrieval. By comparing $\mathbf{V}_t$ with the violence-related terms in the knowledge base, we identify the most relevant violence-related definition $\mathbf{D}_t$ associated with each content frame. Specifically, we calculate the similarity between the visual representations of each frame and the textual definitions.

Given a video frame $\mathbf{I}_t \in \mathbb{R}^{H\times W\times 3}$, the pre-trained Q-Former produces visual queries $\mathbf{V}_t \in \mathbb{R}^{N\times d_\mathrm{q}}$. Here, N is the number of visual queries obtained by the Q-Former, and d_q is the dimension of the visual embedding of each query, respectively. Subsequently, we use the pre-trained text transformer from BLIP2 to obtain the definition embedding $\mathbf{D}_t \in \mathbb{R}^{M\times d_\mathrm{q}}$. Here M is the number of text tokens and the dimension of the embedding of each token are set to d_q. For each frame, we retrieve the definition from **KB** that has the smallest squared Euclidean (L2) distance to the visual embedding.

In BLIP2 (Li et al., 2023a), to calculate the contrastive loss, they propose to compare the [CLS] token from the text embedding and the visual query with the highest pairwise similarity. However, since our goal is not to align visual and text representations, but rather to select the most relevant violent definition for the video, we use mean pooling to compare the entire visual sentence with the entire definition sentence. Specifically, through the mean-pooling step, we obtain and compare the visual sentence $\mathbf{V}_\mathrm{s}^t \in \mathbb{R}^{d_\mathrm{q}}$ and definition sentence $\mathbf{D}_\mathrm{s}^t \in \mathbb{R}^{d_\mathrm{q}}$. After processing the top k frames, we select the definition with the highest pairwise similarity across all frames as the most relevant for the video. We denote the definition sentence vector of the selected definition as $\mathbf{D}_\mathrm{s}$.

Reordering Visual Queries. The 'lost in the middle' effect is a phenomenon observed in LLMs where the model's performance degrades when critical information is situated in the middle of a long input context (Liu et al., 2024). This phenomenon occurs because LLMs tend to focus more on the input's beginning and end, often neglecting the information in the middle. In our method, this effect poses a significant challenge.

To mitigate the 'lost in the middle' effect, we propose reordering the visual queries to prioritize those that are more relevant to the violence definition. By aligning the visual queries ($\mathbf{V}_t$) with the violence definition ($\mathbf{D}_\mathrm{s}$), the LLM receives a coherent and contextually appropriate set of inputs. This alignment ensures that the most critical information is positioned at the beginning of each frame, where the LLM is more likely to focus its attention.

By doing so, the LLM can better understand whether the visual content is directly related to the definition of violence, enabling it to disregard irrelevant definitions when necessary. Overall, this approach not only addresses the 'lost in the middle' effect but also enhances the system's robustness across diverse scenarios.

To reorder the visual queries, first, we compute a similarity vector $\mathbf{S}$ by calculating the dot product between $\mathbf{V}_t$ and $\mathbf{D}_s$ as follows:

$$\mathbf{S} = (\mathbf{V}_1 \cdot \mathbf{D}_s, \ldots, \mathbf{V}_t \cdot \mathbf{D}_s). \tag{5}$$

This similarity vector indicates how closely each visual query aligns with the definition. Next, we sort the indices of the visual queries based on their similarity scores in descending order. This sorting arranges the visual queries from the most to the least relevant according to their similarity to the definition.

Finally, the visual queries are reordered using these sorted indices, resulting in the reordered visual queries $\mathbf{V}_t'$. This process is repeated for each of the top k frames to ensure that the LLM receives a consistent and contextually relevant input.

4.3.3 Output Generator

To create the final prompt for the LLM, we extract the tokens from the user query $\mathbf{T}_\mathrm{u}$ and the most relevant definition $\mathbf{T}_\mathrm{d}$, obtained directly by the LLM tokenizer. The text embeddings are used solely for comparison purposes between visual and text data: first, to obtain the most relevant violence-related definition for the video, and second, to reorder visual queries according to this definition.

Finally, we receive from the Modality Translator module $\mathbf{T}_\mathrm{v}$ for the corresponding frame. We then combine $\mathbf{T}_\mathrm{u}$, $\mathbf{T}_\mathrm{v}$, and $\mathbf{T}_\mathrm{d}$ to form the **final_prompt**: "Is this video: `<frame_1>`, . . . , `<frame_k>` violent, considering that `<definition>` is also an expression of violence?" This prompt enhances LLM's understanding of the concept of violence, helping to mitigate biases and ensure robustness across diverse scenarios. Figure 5 illustrates an example using a single frame.

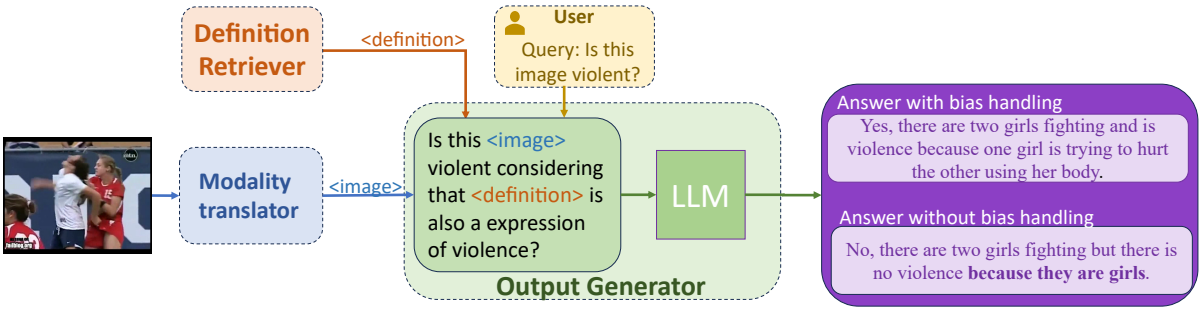


Figure 5: Example of handling bias.

5 EXPERIMENTS

5.1 Datasets

Table 1 summarizes the five datasets we used in the experiments.

Hockey Fight Detection Dataset (Bermejo et al., 2011) was specifically created to introduce a new video dataset for evaluating violence detection systems, where both normal and violent activities occur in similar, dynamic settings. It contains 1 000 clips of action from the National Hockey League (NHL) games. Each clip comprises 50 frames of 720×576 pixels and is manually labeled as "fight" or "non-fight".

Movies Fight Detection Dataset (Bermejo et al., 2011) was constructed to explore the generalization capacity of learning fight patterns. It consists of 200 video clips obtained from action movies, of which 100 contain fight, and 100 videos with non-fight scenes from football games, and other events. The dataset contains videos depicting a wide variety of scenes that were captured at different resolutions and are manually labeled as "fight" or "non-fight".

Surveillance Camera Fight Dataset (Aktı et al., 2019) was collected mostly from YouTube. It encompasses various fight scenarios, including kicks, fists, hitting with objects, and wrestling. The dataset features footage from different locations such as cafes, bars, streets, buses, and shops. In total, there are 300 videos in the dataset: 150 depict fighting sequences, while the remaining 150 show non-fight sequences. These videos vary in resolution, and frame numbers, and have an approximate duration of two seconds each.

Real-World Fighting-2000 Dataset (RWF-2000) (Cheng et al., 2021) was collected from YouTube. It consists of 2 000 video clips captured from around 1 000 raw surveillance videos with extended footage. This dataset is split into two parts: the training set (80%) and the test set (20%), in which half depicts violent behaviors, while the other half represents non-violent activities. The videos have different resolutions and are uniformly cut into 5-second clips with a frame rate of 30 frames per second.

XD-Violence Dataset (Wu et al., 2020) is a large-scale and multi-scene dataset that was collected from movies and YouTube. The dataset includes a total of 91 movies, which were used to collect both violent and non-violent events. Additionally, in-the-wild videos were collected from YouTube. This dataset has a total duration of 217 hours, containing 4 754 untrimmed videos with audio signals and weak labels, from which 2 405 are classified as violent, while 2 349 are non-violent. The dataset is split into two parts: a training set containing 3 954 videos and a test set with 800 videos. The test set consists of 500 violent videos and 300 non-violent videos.

5.2 Implementation Details

In the Key-frame selection step, we utilized the Gunnar Farneback technique, which is implemented in the Open Source Computer Vision Library (OpenCV)[1] to calculate dense optical flow. We set the parameter $k = 5$, meaning that five frames are selected to represent each video. This selection helps in reducing the computational load while maintaining essential motion information.

We employed a divide-and-conquer strategy for longer videos, which we define as those containing more than 5 000 frames. This approach involves splitting the video into up to five parts, treating each part independently to manage the data more efficiently. A video is classified as violent if at least one of its parts contains violent content.

For the VLM (modality translator), the architecture includes a Q-Former with $N = 32$ visual queries, each having a dimension of $d_q = 768$ (Li et al., 2023a). Additionally, the VLM has an FC layer that linearly projects each output visual query embedding

[1] https://opencv.org/

Table 1: Dataset description.

Dataset	*Data Scale* (# clips)	Length/Clip	# Violent clips	# Non-violent clips	Source of scenarios
Movies	200	1.6-2 sec	100	100	Movies and sports
Surveillance Fight	300	1.4-2 sec	150	150	CCTV and mobile cameras
RWF-2000[2]	400	5 sec	200	200	CCTV and mobile cameras
Hockey	1000	1.6-1.96 sec	500	500	Ice hockey games
XD-Violence[2]	800	0-10 min	500	300	Movies, sports, games, hand-held cameras, CCTV, car cameras, etc.

to match the text embedding dimension of the LLM, resulting in a 2048-dimensional output layer.

For the LLM, we chose the instruction fine-tuned version of Google's T5 (Raffel et al., 2020), also known as the Text-to-Text Transfer Transformer Language Model (Flan-T5) (Chung et al., 2024). Since we intend for VIVID to be used in detecting violence in real-time scenarios, we selected the Flan-T5-XL (Chung et al., 2024), which contains around 3 billion parameters. This model size allows for a nuanced understanding and processing of complex language inputs on a single NVIDIA GeForce RTX 3090 GPU, which has 24 GB of GDDR6X memory, and a boost clock of 1 700 MHz .

5.3 Experimental Results

5.3.1 Comparison with LLM-Based Methods

We compare VIVID with other LLM-based methods using the test split of video clips from each dataset as input, along with the user prompt: "Is this video violent?" This user query is consistently applied across all baseline methods to ensure a fair comparison.

The results of our experiments, as summarized in Table 2 , reveal significant insights into the performance of various models on the five datasets. VIVID consistently outperforms other LLM-based models across most datasets, achieving improvements of 0.01–0.324, 0.007–0.187, 0.014–0.294, and 0.066–0.334 in terms of accuracy, and 0.01–0.492, 0.009–0.446, 0.011–0.277, and 0.062–0.537 in terms of F1-Score on the Movies, RWF-2000, Hockey, and XD-Violence datasets, respectively.

In the Movies dataset, VIVID significantly outperforms other LLM-based models by effectively capturing complex visual and contextual cues. It also leads in the RWF-2000 dataset, demonstrating its effectiveness in real-world fight detection. The Hockey dataset results show substantial improvements, highlighting VIVID's capability to detect violent actions in fast-moving and complex interactions. In the XD-Violence dataset, VIVID's robustness is validated by its ability to handle diverse and complex violent scenarios.

The effectiveness of our method lies in its ability to identify potential violent frames and manage various biases associated with LLMs: inherited bias, guardrails bias, and alignment bias. Figure 7 shows the classification made by VIVID over the examples shown in Figures 1c, 2, and 3b, where our method successfully identifies the videos as violent. As illustrated in Figure 8, incorporating a violence-related term in the prompt enables users to address the bias and influence the LLM's judgment based on the definition of violence. The dataset (Bermejo et al., 2011) categorizes all martial arts as violent sports. Therefore, by including martial arts as violence-related terms in the **KB**, the LLM's response aligns with the definition of Karate as a violent sport. This comprehensive approach ensures that our method remains robust and effective across various scenarios, providing a reliable tool for bias management in LLMs.

In the Surveillance Fight dataset, the "InstructBLIP" model shows the highest accuracy 0.796, while the "LLaMA-VID" model achieves the highest F1-Score 0.805. Our model performs well with an accuracy of 0.736 and an F1-Score of 0.781 but does not lead in this dataset. This fact suggests that while our model is effective, it may struggle with the specific challenges posed by surveillance footage, such as varying lighting conditions like insufficient illumination, and occlusions as shown in Figure 6. These challenges could be addressed by jointly fine-tuning the VLM and the LLM with a small amount of violent and non-violent videos under these conditions.

[2]The Data Scale column lists video clips from the test split of the dataset, used for model comparison. We use only the test split to align our evaluation with other methods that measure performance using the test split.

Table 2: Comparison with LLM-based methods in terms of Accuracy and F1-Score.

Model	LLM Backbone	Movies		Surveillance Fight		RWF-2000		Hockey		XD-Violence	
		Acc	*F1*	*Acc*	*F1*	*Acc*	*F1*	*Acc*	*F1*	*Acc*	*F1*
BLIP2[3]	Flan-T5-XL (3B)	0.900	0.880	0.776	0.797	0.767	0.760	0.841	0.859	0.760	0.770
InstructBLIP[3]	Vicuna-1.1 (7B)	0.810	0.765	**0.796**	0.784	0.742	0.685	0.932	0.932	0.535	0.436
X-InstructBLIP	Vicuna-1.1 (7B)	0.880	0.863	0.793	0.800	0.785	0.766	0.856	0.831	0.648	0.616
Video-LLaMA	LLaMA-2 (7B)	0.661	0.492	0.656	0.511	0.610	0.365	0.660	0.676	0.492	0.330
Video-LLaVA	Vicuna-1.5 (7B)	0.975	0.974	0.726	0.777	0.782	0.802	0.662	0.743	0.726	0.802
LLaMA-VID	Vicuna-1.5 (7B)	0.965	0.963	0.783	**0.805**	0.790	0.794	0.940	0.942	0.755	0.805
VIVID	Flan-T5-XL (3B)	**0.985**	**0.984**	0.736	0.781	**0.797**	**0.811**	**0.954**	**0.953**	**0.826**	**0.867**

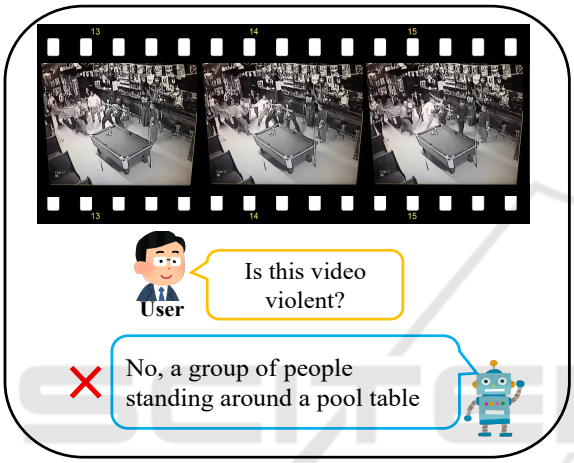


Figure 6: Example where occlusion caused by the pool table lights partially blocking the violent scene, along with lack of illumination, resulted in the LLM failing to detect the violent scene correctly.

However, the primary objective of our research is to evaluate the capabilities of an LLM to classify violent content in a zero-shot manner. This approach allows us to assess the inherent strengths and limitations of the LLM without further modifications. Nonetheless, we recognize the potential for improvement through fine-tuning and consider it a promising direction for our future work.

5.3.2 Comparison with Traditional Hand-Crafted Features and Deep Learning-Based Methods

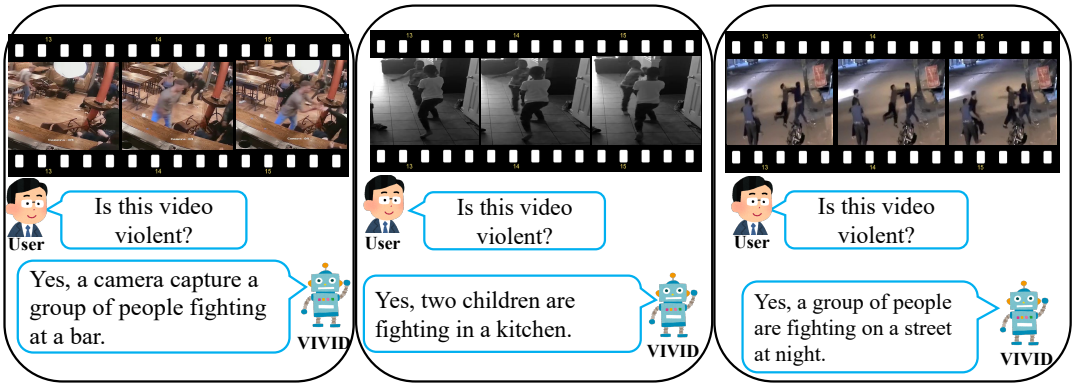
When comparing VIVID with traditional hand-crafted features and deep learning-based methods, as shown in Table 3, we notice a clear difference in performance. Traditional methods such as HOG + HIK (Bermejo et al., 2011), HOF + HIK (Bermejo et al., 2011), and MoSIFT + HIK (Bermejo et al., 2011) show lower accuracy across the datasets. For instance, MoSIFT + HIK achieves an accuracy of 0.895 in the Movies dataset and 0.909 in the Hockey dataset, which are significantly lower than our model's performance. This fact indicates that traditional hand-crafted features are less effective in capturing complex patterns of violence in videos. Deep learning-based methods, such as Xception + Bi-LSTM + attention (Aktı et al., 2019), Flow Gated (Cheng et al., 2021), and Conv3D (Park et al., 2024), demonstrate high performance. For example, Xception + Bi-LSTM + attention achieves an accuracy of 1.0 in the Movies dataset and 0.98 in the Hockey dataset. However, these methods often require a considerable amount of training data to generalize. Due to the restricted amount of available training data, the resulting models are often ad-hoc and fail to generalize violent behavior across different scenarios.

In contrast, VIVID uses zero-shot classification, which leverages the pre-existing knowledge of the LLM along with bias handling to enhance performance. This approach not only effectively classifies violent content but also provides interpretability. The interpretability feature is particularly advantageous as it helps users understand why a video instance was classified as violent or not, making it a valuable tool for real-world applications. Even though VIVID does not always achieve the highest accuracy compared to some deep learning methods, its performance is competitive or comparable to these methods. Additionally, our method outperforms the Xception + Bi-LSTM + attention method in the Surveillance Fight dataset, achieving an accuracy of 0.736 compared to 0.720.

5.3.3 Ablation Study

To investigate the effects of reordering visual queries, we conducted experiments using three distinct vari-

[3]BLIP2 and InstructBLIP do not support video input. We analyze violent content at the frame-level, extracting frames as in X-InstructBLIP. If any frame contains violent content, the entire video is classified as violent.



(a) BLIP-2. (b) Video-LLaMA. (c) Video-LLaVA.

Figure 7: Example in which VIVID successfully handles the biases of the LLMs.

Table 3: Comparison with hand-crafted features and deep learning-based methods in terms of accuracy. A hyphen indicates that the results are not provided in the corresponding paper.

Type	Model	Movies	S. Fight	RWF 2000	Hockey
Hand Crafted Features	HOG + HIK	0.490	-	-	0.917
	HOF + HIK	0.590	-	-	0.886
	MoSIFT + HIK	0.895	-	-	0.909
DL Based	Xception + Bi-LSTM + attention	1.000	0.720	-	0.980
	Flow Gated	1.000	-	0.872	0.980
	Conv3D	1.000	-	-	0.981
LLM based	VIVID	0.985	**0.736**	0.797	0.954

ants of our method. These variants share the common approach of comparing a visual sentence with text embedding to identify the most relevant violence-related definition for an image. However, in each variation, the text and visual embeddings interact differently.

Variant 1: Inspired by BLIP2 (Li et al., 2023a), we compare the [CLS] token from the text with the visual queries. The most relevant violent definition is identified as the one with the highest pairwise similarity.

Variant 2: Instead of using the [CLS] token, we use mean pooling. Specifically, we compare a visual sentence obtained by mean pooling all the visual queries with a definition sentence obtained by mean pooling all sentence tokens, as explained in Section 4.3.2.

VIVID: In addition to using the mean pooling approach from Variant 2, we also reorder the visual queries. By calculating the dot product between the visual sentence and each text token, we can prioritize visual queries that are more relevant to the violence definition, as explained in Section 4.3.2.

Table 4 shows the results of each variant on the five datasets. From these results, it is evident that

Table 4: Comparison of our method's variants in terms of Accuracy.

Model	Movies	S. Fight	RWF 2000	Hockey	XD Violence
Variant 1	0.915	0.706	**0.800**	0.942	0.671
Variant 2	0.940	0.680	0.792	0.920	0.730
VIVID	**0.985**	**0.736**	0.797	**0.954**	**0.826**

VIVID demonstrates superior performance across most datasets. It achieves the highest accuracy in the Movies, Surveillance Fight, Hockey, and XD-Violence datasets. Although VIVID was not the best in the RWF-2000 dataset, the difference in accuracy was small enough to be negligible. Therefore, we consider VIVID to have achieved same performance to Variant 1 in this dataset. Overall, the robust performance of VIVID in four out of five datasets highlights its effectiveness in enhancing the model's accuracy through the reordering of visual queries.

6 CONCLUSIONS

In this paper, we introduced VIVID, a novel method for detecting violent content in videos by integrating Vision-Language Models (VLM) with Large Language Models (LLMs). Our method demonstrates superior performance across multiple datasets, effectively capturing complex visual and contextual cues.

VIVID's zero-shot classification capability allows it to identify violent content without additional training, making it an alternative in scenarios with limited labeled data. By incorporating an external knowledge base, VIVID mitigates biases and provides reliable results. Its interpretability characteristics enhance user trust by explaining the reasons behind its classifications.

There are a number of directions that can be ex-

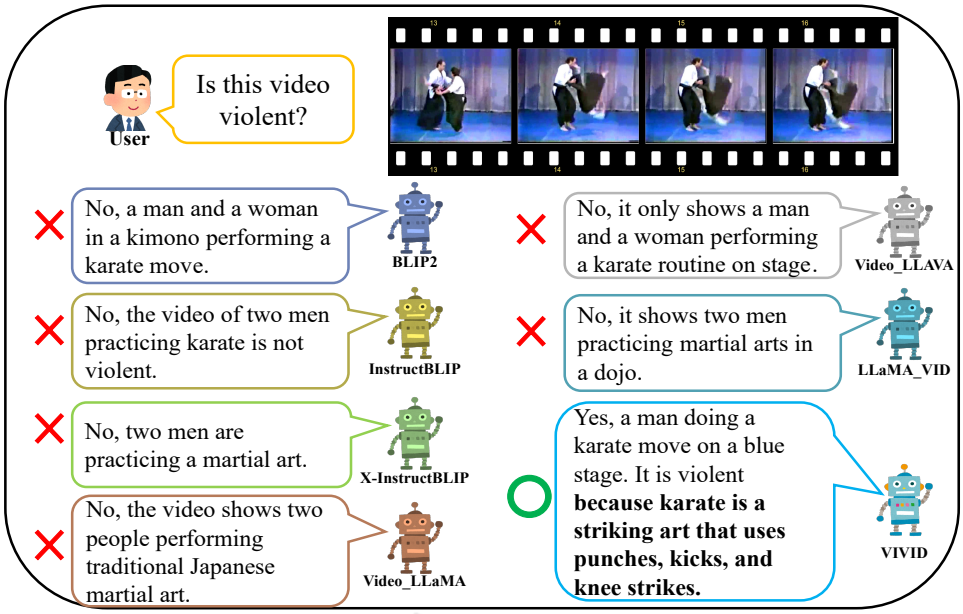


Figure 8: Example in which the inclusion of a violent-related definition improves performance compared to the baseline methods.

plored for future research, including (1) fine-tuning the VLM and LLM with a small amount of violent and non-violent videos to address specific challenges like varying lighting conditions and occlusions in surveillance footage, (2) including prompt engineering to have a broader range of violence-related terms and scenarios, and (3) enhancing the interpretability features to provide more detailed explanations for the model's classifications. Overall, VIVID offers a balanced approach combining accuracy, interpretability, and generalizability, making it a robust solution for violent content classification in real-world applications.

REFERENCES

Abdali, A.-M. R. and Al-Tuma, R. F. (2019). Robust Real-Time Violence Detection in Video Using CNN and LSTM. In *Proc. IEEE SCCS*, pages 104–108.

Aktı, Ş., Tataroğlu, G. A., and Ekenel, H. K. (2019). Vision-Based Fight Detection from Surveillance Cameras. In *Proc. IEEE IPTA*, pages 1–6.

American Psychological Association (2024). Physical abuse and violence. Accesed on 20/12/204 from: https://www.apa.org/topics/physical-abuse-violence.

Bermejo, E. N., Déniz, O. S., Bueno, G. G., and Sukthankar, R. (2011). Violence Detection in Video Using Computer Vision Techniques. In *Proc. CAIP, Part II 14*, pages 332–339.

Bobick, A. F. and Davis, J. W. (2001). The Recognition of Human Movement Using Temporal Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267.

Cheng, M., Cai, K., and Li, M. (2021). RWF-2000: An Open Large Scale Video Database for Violence Detection. In *Proc. ICPR*, pages 4183–4190.

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., Li, Y., Wang, X., Dehghani, M., Brahma, S., et al. (2024). Scaling Instruction-Finetuned Language Models. *Journal of Machine Learning Research*, 25(70):1–53.

Dai, W., Li, J., Li, D., Tiong, A. M. H., Zhao, J., Wang, W., Li, B., Fung, P., and Hoi, S. (2023). InstructBLIP: Towards General-Purpose Vision-Language Models with Instruction Tuning. *arXiv preprint arXiv:2305.06500*.

Dalal, N., Triggs, B., and Schmid, C. (2006). Human Detection Using Oriented Histograms of Flow and Appearance. In *Proc. ECCV, Part II 9*, pages 428–441.

Das, S., Sarker, A., and Mahmud, T. (2019). Violence Detection from Videos Using HOG Features. In *Proc. IEEE EICT*, pages 1–5.

De Souza, F. D., Chavez, G. C., do Valle Jr, E. A., and Araújo, A. d. A. (2010). Violence Detection in Video Using Spatio-Temporal Features. In *Proc. SIBGRAPI*, pages 224–230.

Deniz, O., Serrano, I., Bueno, G., and Kim, T.-K. (2014). Fast Violence Detection in Video. In *Proc. VISIGRAPP (VISAPP)*, volume 2, pages 478–485.

Dong, Y., Mu, R., Zhang, Y., Sun, S., Zhang, T., Wu, C., Jin, G., Qi, Y., Hu, J., Meng, J., et al. (2024). Safeguarding Large Language Models: A Survey. *arXiv preprint arXiv:2406.02622*.

Febin, I., Jayasree, K., and Joy, P. T. (2020). Violence Detection in Videos for an Intelligent Surveillance System Using MoBSIFT and Movement Filtering Algo-

rithm. *Pattern Analysis and Applications*, 23(2):611–623.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., and Wang, H. (2023). Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv preprint arXiv:2312.10997*.

Hassner, T., Itcher, Y., and Kliper-Gross, O. (2012). Violent Flows: Real-Time Detection of Violent Crowd Behavior. In *Proc. IEEE CVPR Workshops*, pages 1–6.

Herrenkohl, T. I. (2011). *Violence in Context: Current Evidence on Risk, Protection, and Prevention*. OUP USA.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Li, J., Li, D., Savarese, S., and Hoi, S. (2023a). BLIP-2: Bootstrapping Language-Image Pre-Training with Frozen Image Encoders and Large Language Models. In *Proc. ICML*, pages 19730–19742.

Li, Y., Wang, C., and Jia, J. (2023b). LLaMA-VID: An Image is Worth 2 Tokens in Large Language Models. *arXiv preprint arXiv:2311.17043*.

Lin, B., Zhu, B., Ye, Y., Ning, M., Jin, P., and Yuan, L. (2023). Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. *arXiv preprint arXiv:2311.10122*.

Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., and Liang, P. (2024). Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Mumtaz, N., Ejaz, N., Habib, S., Mohsin, S. M., Tiwari, P., Band, S. S., and Kumar, N. (2023). An Overview of Violence Detection Techniques: Current Challenges and Future Directions. *Artificial Intelligence Review*, 56(5):4641–4666.

Panagopoulou, A., Xue, L., Yu, N., Li, J., Li, D., Joty, S., Xu, R., Savarese, S., Xiong, C., and Niebles, J. C. (2023). X-InstructBLIP: A Framework for Aligning X-Modal Instruction-Aware Representations to LLMs and Emergent Cross-Modal Reasoning. *arXiv preprint arXiv:2311.18799*.

Park, J.-H., Mahmoud, M., and Kang, H.-S. (2024). Conv3D-Based Video Violence Detection Network Using Optical Flow and RGB Data. *Sensors*, 24(2):317.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Senst, T., Eiselein, V., Kuhn, A., and Sikora, T. (2017). Crowd Violence Detection Using Global Motion-Compensated Lagrangian Features and Scale-Sensitive Video-Level Representation. *IEEE Transactions on Information Forensics and Security*, 12(12):2945–2956.

Sudhakaran, S. and Lanz, O. (2017). Learning to Detect Violent Videos Using Convolutional Long Short-Term Memory. In *Proc. IEEE AVSS*, pages 1–6.

Sun, Z., Ke, Q., Rahmani, H., Bennamoun, M., Wang, G., and Liu, J. (2022). Human Action Recognition from Various Data Modalities: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3200–3225.

Szeliski, R. (2022). *Computer Vision: Algorithms and Applications*. Springer Nature.

Traoré, A. and Akhloufi, M. A. (2020). Violence Detection in Videos Using Deep Recurrent and Convolutional Neural Networks. In *Proc. IEEE SMC*, pages 154–159.

Ullah, F. U. M., Obaidat, M. S., Ullah, A., Muhammad, K., Hijji, M., and Baik, S. W. (2023). A Comprehensive Review on Vision-Based Violence Detection in Surveillance Videos. *ACM Comput. Surv.*, 55(10).

Ullah, F. U. M., Ullah, A., Muhammad, K., Haq, I. U., and Baik, S. W. (2019). Violence Detection Using Spatiotemporal Features with 3D Convolutional Neural Network. *Sensors*, 19(11):2472.

Wang, H., Kläser, A., Schmid, C., and Liu, C.-L. (2013). Dense Trajectories and Motion Boundary Descriptors for Action Recognition. *International Journal of Computer Vision*, 103:60–79.

Wu, P., Liu, J., Shi, Y., Sun, Y., Shao, F., Wu, Z., and Yang, Z. (2020). Not Only Look, But Also Listen: Learning Multimodal Violence Detection Under Weak Supervision. In *Proc. ECCV*, pages 322–339. Springer.

Zhang, H., Li, X., and Bing, L. (2023). Video-LLaMA: An Instruction-Tuned Audio-Visual Language Model for Video Understanding. *arXiv preprint arXiv:2306.02858*.