# VPLLM: Next Frame Video Prediction with LLMs

Ziv Fenigstein
Ben Gurion University of the Negev
Beer Sheba, Israel
zivfenig@post.bgu.ac.il

Tal Klein
Ben Gurion University of the Negev
Beer Sheba, Israel
tak@post.bgu.ac.il

Roi Garber
Ben Gurion University of the Negev
Beer Sheba, Israel
roigar@post.bgu.ac.il

Eliya Aharon
Ben Gurion University of the Negev
Beer Sheba, Israel
eliyaah@post.bgu.ac.il

Ruben Chocron
Ben Gurion University of the Negev
Beer Sheba, Israel
rubencho@post.bgu.ac.il

## ABSTRACT

This paper explores the efficacy of Large Language Models (LLMs) in next-frame video prediction, a domain that traditionally relies on extensive training of specialized models. By implementing a novel, training-free pipeline, we leverage pre-existing LLMs to predict video frames without the need to train models from scratch. Our methodology employs a combination of image-to-text transformations and predictive modeling using various configurations of LLMs and image processing tools. We compare the performance of different setups including BLIP2, YOLO, and ChatGPT-4, both in generating detailed frame descriptions and predicting subsequent video frames. The results indicate that the combination of BLIP2 with YOLO for feature enhancement and ChatGPT-4 for generating descriptive texts performs optimally in producing accurate and detailed frame descriptions. However, for the direct prediction of the next video frame, the simpler setup of just BLIP2 was found to be slightly more effective, although the differences among the models were marginal. These findings highlight the potential of using existing LLMs for video prediction tasks, offering a more accessible alternative to traditional methods that require heavy computational investments and extensive training.

## KEYWORDS

Video prediction, language-vision model, multimodal AI, Large Language Models (LLMs), image-to-text models, computer vision

**ACM Reference Format:**
Ziv Fenigstein, Tal Klein, Roi Garber, Eliya Aharon, and Ruben Chocron. 2025. VPLLM: Next Frame Video Prediction with LLMs . In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 9 pages.

## 1 INTRODUCTION

Video prediction, particularly the prediction of subsequent frames in a video sequence, stands as a cornerstone challenge in the field of computer vision and artificial intelligence. Traditional approaches to this problem have largely depended on extensive model training, involving deep learning architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and more recently, transformer-based models. While these methods are powerful, they often require significant computational resources and expertise for training them model which can be a barrier to entry for many researchers and practitioners.

Recent advancements in Large Language Models (LLMs) have opened new avenues for addressing complex tasks across various domains without the need for domain-specific model training. LLMs, known for their flexibility and robustness, have demonstrated remarkable success in natural language processing and have increasingly been applied to multimodal tasks that include both text and visual components. This research seeks to harness the capabilities of LLMs to innovate within the realm of video prediction, proposing a novel approach that leverages existing pre-trained models to predict future video frames.

The primary motivation behind this study is to explore whether LLMs can effectively understand and generate visual content with the same efficacy they have shown with textual data. By utilizing models such as BLIP2, YOLO for object detection and enhancement, and ChatGPT-4 for generating rich, contextual descriptions, this research aims to create a pipeline that bypasses the traditional requirements for heavy computational training. This approach not only addresses the practical limitations posed by resource-intensive training processes but also aligns with a growing shift towards more sustainable AI practices that emphasize the reuse and adaptation of existing models over building new ones from scratch.

Furthermore, this study is situated within a broader effort to bridge the gap between language and vision, reflecting on how language models interpret and generate visual content. The integration of LLMs in video prediction challenges the conventional boundaries of what these models can achieve and sets the stage for future innovations where language and vision converge seamlessly.

This paper will detail the methodology of using a combination of LLMs and visual processing tools to predict video frames, evaluate the performance of various configurations, and discuss the implications of these findings. We hypothesize that LLMs can match, and potentially exceed the performance of traditional video prediction models in certain contexts, whilst offering a simpler and more accessible approach. The outcomes of this research could pave the way for new applications in video analysis, content generation, and beyond, where ease of use and efficiency are paramount. In the following sections, we will explore related works that have laid the groundwork for this study, describe the methodology in detail, present our empirical evaluations, and discuss the broader implications of our findings within the fields of artificial intelligence and computer vision.

*Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), A. El Fallah Seghrouchni, Y. Vorobeychik, S. Das, A. Nowe (eds.), May 19 – 23, 2025, Detroit, Michigan, USA.*

## 2 RELATED WORK

Previous research combining computer vision and LLMs has focused mainly on computer understanding of videos, and less about prediction. Many works focus on the aspect of temporal and spatial understanding with the help of LLMs. Others focus on utilizing LLMs to better caption and describe videos. Still, there exist challenges in the efficiency of such tasks and the alignment between video, image, and text in these multi-modal models.

Valley [12] attempts to create a foundational model with multi-model capabilities, accepting text, images, and video altogether. The framework combines modules designed for temporal understanding, vision, and alignment between visual and textual information. First pre-training a projection module for video-to-text transitions, the model is then fine-tuned on a variety of other tasks such as video description and task recognition. Valley achieved state-of-the-art performance on benchmarks like MSVD-QA, MSRVTT-QA, and ActivityNet-QA. These are known benchmarks in the domain of video prediction, yet it is important to note that our work utilizes and evaluates different datasets.

In another work in the domain of mixing video and language models, SEER [5] puts in place a framework in which the next frame is predicted, given a few frames and an accompanying instruction text. For example, given an initial frame of a beer can, and the instruction “tipping a beer can,” the output video would be that same can being tipped over. This model can be seen as an extension of text-to-image models. SEER outperformed some state-of-the-art models such as CogVideo, after training for a quarter of the training time for CogVideo.

In a significant contribution to computer video understanding, ST-LLM [11] shows that LLMs are a good tool to use for sequence modeling in videos. That is, given spatial- temporal video tokens, a LLM can accurately learn what is happening in the video and in what sequence. However, longer videos may lead to high computational costs, and LLMs may struggle in finding an appropriate context length and therefore output “hallucinations”. To avoid this, ST-LLM implements a combination of masking the input,leading to smaller and more constant input sizes, as well as choosing a subset of relevant frames to reduce computations. ST-LLM achieved state-of-the-art results on multiple benchmarks.

In another related work, VTimeLLM [6] focuses specifically on the temporal comprehension aspect of video LLMs. Given a video, many do not fully possess the capacity to precisely say when an event started and ended. In their framework, they adopt a three stage training process to enhance the temporal understanding of the LLM: “image-text pairs for feature alignment, multiple-event videos to increase temporal boundary awareness, and high-quality video-instruction tuning to further improve temporal understanding ability.” [6]. VTimeLLM brings remarkable results in video time comprehension tasks, better than various other video LLMs.

In the video summarization domain, in particular action recognition, is the SMART framework [4]. This method tackles the high processing cost by analyzing frames in batches and using attention mechanisms to identify only the most relevant frames. This algorithm brings efficiency to the table by reducing the amount of frames to be processed, speeding up the classification of actions in videos by ’4 to 10 times.’ [4]

In regards to image captioning, most models manage to provide simple and short captions. For many tasks, these descriptions are too short and not detailed enough. This research [1] provides a solution to output more comprehensive captions by utilizing LLMs to combine image captions created by multiple state-of-the-art models. As a result of this fusion, this method creates longer and more descriptive captions of a given image. In a framework like ours, highly detailed captions of an image may be helpful for a LLM to predict the next frame.

## 3 BACKGROUND

Video prediction, and in particular next-frame prediction, has been researched quite extensively in the past few years. The accepted methods today for predicting the next frame of a video differ between models such as recurrent neural networks (RNNs), convolutional neural networks (CNNs), long short term memory networks (LSTM), and transformer based architectures. In this section we will cover state-of-the-art methods and explain crucial aspects for video prediction, and highlight some of the challenges this domain is facing.

### 3.1 Existing video prediction methods

The ConvLSTM paper [2] describes a state-of-the-art approach to next frame prediction, using a combination of convolutional neural networks and LSTMs to better ‘remember’ the content of the previous frames. Later works extended this research, using Generative Adversarial Networks (GANs) [9] combined with CNNs, LSTMs, and Convolutional LSTMs (a continuation of the ConvLSTM framework) for next frame prediction. In a more recent study and as a continuation of the ConvLSTM works, this article [15] presents a novel set of transformer models for video predictions. This work aims to reduce the complexity of current Transformers and does so by regulating the attention mechanisms it uses. Moreover, this model proposes three different models to predict video, an autoregressive, a partially autoregressive, and a non-autoregressive model, each providing the advantage of inference quality or inference speed, respectively. These models achieved similar or higher performance than convolutional LSTMs while improving computational efficiency. Most video prediction models are based on complex architectures, including stacked RNNs or Transformers. While the performance of these models is considerable, they are not scalable to larger datasets. In an attempt to reduce this complexity, SimVP [3] proposes a simpler and more scalable approach to video prediction algorithms, with an end-to-end trained encoder-decoder.

### 3.2 Frame selection

In terms of machine video summarization, there exists a serious challenge - the expensive and extensive data labeling process. People may have different subjective views of an accurate video summarization, hence requiring the data labeling to also be personalized. A possible solution to this problem is **frame selection**, aimed at identifying the most significant frames within a video to summarize or represent its content effectively. This task is essential because it allows for efficient content management, quicker data processing, and less bandwidth usage, which are particularly important in environments with limited resources. By selecting key frames

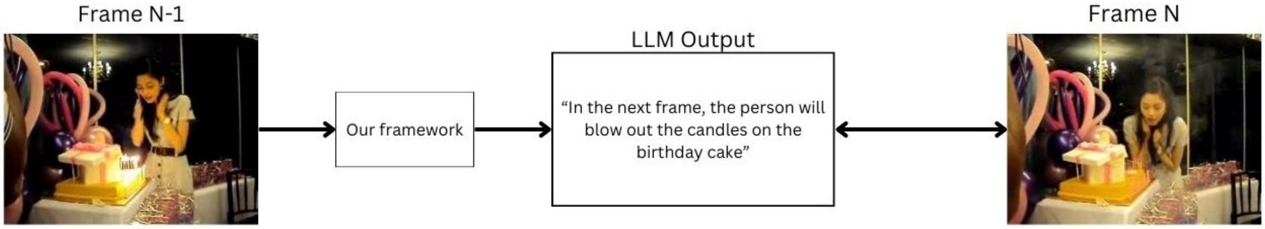
Frame N-1
Our framework
LLM Output
“In the next frame, the person will blow out the candles on the birthday cake”
Frame N

**Figure 1: Example of frame prediction**

that accurately reflect the content of a video, analysts and algorithms can perform more efficient analyses without the need for processing the entire video stream. As an example of advancements in this field, Huang and Wang [7] developed a key-frames selection framework using deep learning to optimize video summarization. Their approach incorporates Capsules Net and self-attention mechanisms, highlighting the ongoing innovations that aim to refine the accuracy and efficiency of frame selection techniques. This work [8] proposes an unsupervised method to summarize videos through feature extraction and clustering for frame selection (disregarding redundant frames), creating a condensed video sequence.

### 3.3 Feature Selection in Images

Feature extraction is a critical process in image processing and computer vision that involves identifying and analyzing specific attributes or patterns within an image. This step is essential for transforming raw data into a format that can be effectively used for further analysis or machine learning tasks. In the realm of object detection, feature extraction serves as the backbone for identifying relevant objects within various scenes with speed and accuracy. A significant advancement in this domain was achieved with the development of the You Only Look Once (YOLO) architecture [14], which revolutionized real-time object detection systems. YOLO uniquely combines feature extraction and object detection into a single convolutional network that processes the full image, predicting multiple bounding boxes and class probabilities for those boxes simultaneously. This approach allows for extremely fast object detection rates, making it ideal for applications requiring real-time processing. The method’s effectiveness in extracting meaningful features from images have set a new standard in object detection technologies [14]. In our work, we use this method to aid the LLMs better understand what is going on in a given frame.

## 4 METHODOLOGY

In this study, we propose a training-free pipeline that leverages large language models (LLMs) to analyze and predict video frame sequences without requiring intensive computational resources or pre-trained models. This novel methodology circumvents the need for traditional training-based pipelines, relying entirely on existing tools, making it computationally efficient and accessible.

As shown in Figure 2, the pipeline begins with a video divided into N-1 individual frames, where N-1 represents the batch size. The batch size determines the number of frames used for predicting frame N based on the N-1 preceding frames. Each of these preceding frames is processed individually using an image-to-text LLM, guided by a prompt designed to generate a detailed description of each frame. For example, the prompts may look like the following:

Captioning Prompt

""Describe the image in detail by identifying all the visible objects. For each object, include the following information:

(1) Identification: Clearly identify the object (e.g., tree, person, car, animal).
(2) Location: Specify the object’s position within the image (e.g., top-right corner, near the center, in the background).
(3) Appearance: Describe the object’s color, shape, texture, and any distinctive features.
(4) Size and Scale: Provide an estimate of the object’s relative size (e.g., large, small, medium) and how it compares to other objects in the image.
(5) Condition: Explain the object’s state or condition (e.g., clean, rusty, worn, shiny).
(6) Material Composition: Guess the material the object might be made of (e.g., wood, metal, glass).
(7) Relation to Other Objects: Describe how the object interacts with or relates to other objects in the image (e.g., a person holding a book, a chair next to a table).
(8) Surroundings: Mention any nearby objects, features, or context that provide additional details about the object’s environment.
(9) Activity or Movement (if applicable): Indicate if the object is involved in an action or is static (e.g., a dog running, a car parked).
(10) Perspective: Highlight how the camera’s angle or the lighting affects the perception of the object (e.g., shadows, reflections, partial visibility).

Ensure the description is thorough and well-organized, covering every visible detail. Use clear and precise language to create an accurate mental image of the scene.""

These descriptions capture the unique characteristics of each frame while providing the temporal and visual context necessary for frame prediction. Once all frame descriptions are generated, they are concatenated into a single coherent text input, maintaining the temporal context across frames. Temporal context, as highlighted in related work, is critical for frame prediction because it allows the model to capture dynamic changes and relationships between consecutive frames. This approach ensures that predictions of Frame N are informed by a comprehensive narrative derived from prior frames, improving the overall accuracy and reliability of the results. The concatenated descriptions are fed into the LLM with a guiding prompt. The guiding prompt, for a frame window of 2, would be the following:

> Prediction Prompt
>
> ""The following captions describe the frames of a video:
> Frame 1: *{frame_1_description}*
> Frame 2: *{frame_2_description}*
> Based on this sequence, predict the caption for the next frame. Continue the story logically and include details about actions, objects, and changes.""

The LLM generates a detailed textual description of Frame N, capturing the anticipated content of the next frame based on the prior sequence. This predicted textual description can then be used as input to a text-to-image LLM, enabling the generation of a visual representation of the predicted frame. By combining these steps, the proposed pipeline provides a training-free approach to video frame prediction, leveraging the strengths of LLMs.

Consider a video of a person blowing out birthday candles, as shown in Figure 1 . When divided into frames, we observe various objects in the surrounding, including the presents on the table, the cake, and the candles themselves. The textual description for Frame N should include the event of the candles having been blown out, if applicable. For instance, if the previous frames show the person preparing to blow out the candles, the description of Frame N would capture the outcome, such as "The person has blown out the candles." This ensures the prediction aligns with the visual context and narrative of the situation.

# 5 EMPIRICAL EVALUATION

## 5.1 Experimental Setup

This study aims to evaluate the following question: Do large language models (LLMs) have the potential to correctly predict video frames without needing to train them from scratch, and can they do so at an accuracy level on par with existing frame prediction methods?

### 5.1.1 *Dataset*.

We use video clips from the UCF101 dataset, dividing them into frames based on a predefined batch size. The UCF101 dataset consists of 13,320 video clips, each labeled with one of 101 action classes. These classes fall into five categories: Human-Object Interaction, Body-Motion Only, Human-Human Interaction, Playing Musical Instruments, and Sports. The dataset is composed of user-submitted videos featuring camera movements and crowded backdrops. It includes diverse viewpoints, scales, camera motions, and lighting conditions, presenting a challenging benchmark for prediction tasks. As part of the pre-processing, we extracted individual frames from the video clips.

Due to computational constraints, it was impractical to process the entire dataset, necessitating the selection of a representative subset for our experiments. To ensure that our evaluation was both meaningful and efficient, we manually analyzed the dataset to identify a diverse range of topics that would challenge our model while still providing clear, interpretable results. Our goal was to select video clips that contained actions that could be distinctly detected and anticipated by our model, allowing us to rigorously assess its ability to capture relevant features and generate accurate embeddings. Rather than selecting clips arbitrarily, we focused on actions that exhibited clear motion patterns and well-defined temporal structures, which would enable us to determine whether the model could consistently recognize and encode these sequences effectively.

With these criteria in mind, we chose video clips depicting activities such as bowling, weight lifting, gymnastics, and blowing out candles. These actions were selected because they presented unique challenges in terms of motion dynamics, temporal dependencies, and object interactions, making them ideal for evaluating the robustness of our model. Bowling, for example, involves a distinct sequence of movements, including the approach, release, and follow-through, which must be accurately captured by the model to produce meaningful embeddings. Weight lifting, on the other hand, requires an understanding of controlled movements and posture stability, as well as the ability to differentiate between different lifting techniques and phases of the lift. Gymnastics introduces another layer of complexity due to its intricate, high-speed motions and rotational dynamics, challenging the model's ability to maintain consistency in representation across varying postures and orientations. Finally, blowing out candles was selected as a contrasting action that involves minimal body movement but still requires an understanding of facial expressions, breath control, and small-scale interactions with the environment.

By selecting these specific categories, we ensured that our dataset subset was not only computationally feasible but also diverse enough to test the generalizability of our approach. Each chosen action presented a different type of challenge for the model, allowing us to examine whether it could effectively capture essential motion features, adapt to different types of activities, and maintain high cosine similarity scores across varying contexts. This careful selection process helped us balance computational efficiency with the need for a rigorous and comprehensive evaluation, ultimately strengthening the reliability and interpretability of our results.

In addition, the LLMs receive carefully crafted prompts that refine their task-specific outputs, ensuring accurate and contextually relevant results. Some of these prompts were created using OpenAI's ChatGPT, asking it to create a prompt that will elicit the most detailed description of an image frame. Other simpler prompts were written by the team.

### 5.1.2 *Computational Resources*.

We utilized a GPU 4090 for computation and developed the code in Python. The LLM models employed include GPT-4o, a versatile multi-LLM model, and Blip2ForConditionalGeneration, which integrates NLP and image

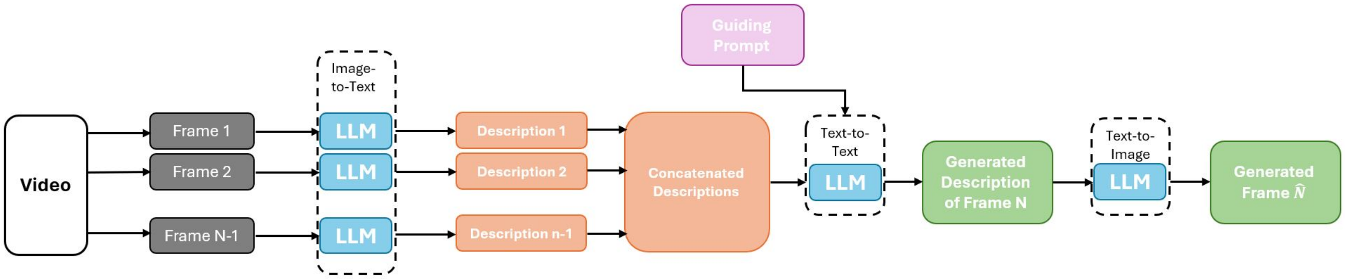

**Figure 2: The VPLLM Pipeline**

processing capabilities to generate textual descriptions. The preprocessing phase involved extracting frames from video clips. For video processing, we utilized libraries such as PIL and OpenCV (cv2) for frame extraction and image handling, Base64 for format conversion to make the data compatible with the LLM, Glob for managing file paths, and Torch for computational operations.

*5.1.3* ***Metrics****.* For description evaluations, we employed a CLIP model to generate an embedding for each image. Additionally, we used a Sentence Transformer model to create an embedding for the corresponding textual description. We then computed the cosine similarity between the two embeddings to assess how well the generated image aligns with its textual description. This method provides a direct measure of the semantic consistency between the generated frame and its description. However, a key limitation arises due to the nature of our dataset; since the video clips are short and depict simple actions, consecutive frames often receive identical textual descriptions, leading to artificially high similarity scores. To mitigate this issue, we introduced for descriptions that exhibit minimal variation across frames. This adjustment ensures that the similarity score more accurately reflects the quality of the generated image-description pairs.

For evaluating predictions, we applied the same scoring method without the penalty, as the score is computed only for the final predicted frame (N), rather than across multiple frames. This allows us to assess the accuracy of the generated frame without interference from repetitive descriptions in prior frames.

*5.1.4* ***Parameters****.* For our experiments, we use a batch size of 26 to predict the 27th frame in the sequence. We test out pipelines that include different language models, with one combination in which they are used on top of each other, one being used for the initial image captioning and the second to refine the prompts. For this purpose, we used BLIP [10], a strong model for vision-language tasks, and ChatGPT 4 [13] through its API. In terms of augmenting, further enhancing, and highlighting the objects, in the image, we test using the YOLO algorithm [14] and an edge detections filter.

*5.1.5* ***Statistical Tests****.* In our research, we conducted multiple experiments to evaluate the effectiveness of our approach by analyzing the cosine similarity between the embeddings generated by our network and the ground truth embeddings. For each experimental setup, we computed the cosine similarity between the predicted and ground truth embeddings to assess how well our model captured the intended representations. Since our study involved multiple experimental setups with varying conditions, it was essential to determine whether the differences in their results were statistically significant. To achieve this, we performed a series of independent *t*-tests to evaluate whether the cosine similarity values obtained across different setups exhibited meaningful differences.

The primary objective of these statistical tests was to assess whether changes in our experimental conditions led to statistically significant variations in cosine similarity, which would indicate an impact on our model's performance. Each *t*-test compared the distributions of cosine similarity scores obtained from two different experimental setups, with the null hypothesis stating that there was no significant difference between their mean cosine similarity scores. In contrast, the alternative hypothesis suggested that the mean cosine similarity scores between the two setups were significantly different, implying that the specific changes in experimental conditions influenced the model's ability to generate embeddings that aligned with the ground truth.

To ensure the validity of our statistical tests, we examined key assumptions underlying the *t*-test, including normality of the cosine similarity distributions and homogeneity of variances across experimental conditions. Given that cosine similarity values are bounded between -1 and 1, their distribution can sometimes be skewed, especially when most similarities cluster around high values.

By applying *t*-tests to compare cosine similarity results across different experimental setups, we were able to quantitatively assess the impact of various factors on our model's performance. If a *t*-test resulted in a statistically significant *p*-value (typically below 0.05), we concluded that the differences between the experimental setups were unlikely to have occurred due to random variation, thereby indicating that the tested modifications had a measurable effect on the quality of the generated embeddings. Conversely, if no significant difference was found, it suggested that the experimental change did not substantially alter the similarity between predicted and ground truth embeddings. This statistical evaluation allowed us to rigorously validate our experimental findings, ensuring that our conclusions were not merely based on observed trends but were supported by empirical evidence.

*5.1.6* ***Ablation Study****.* To evaluate the impact of different processing techniques on the accuracy and detail of frame descriptions, we conducted a focused ablation study. This study addresses the challenges posed by low-resolution frames in the dataset, where subtle visual details crucial for textual description might be overlooked by LLM. Given the variability in video quality within the UCF101 dataset, this configuration is expected to reveal limitations in capturing fine-grained details from the frames. To address this, we tested multiple model configurations:

1. BLIP2 (Baseline): Used as the reference model to generate initial frame descriptions without additional enhancements. This model leverages a Transformer-based architecture to generate frame descriptions directly from raw video frames without any enhancements or preprocessing. By assessing BLIP2's default performance, we can gauge the impact of augmentations on frame description.
2. BLIP2 + Edge Detection: Applied edge detection filters to highlight structural features, such as object boundaries and contours. This augmentation is particularly beneficial for emphasizing shapes and patterns that may be less distinguishable in low-resolution videos. By making structural details more prominent, edge detection aims to improve the LLM's ability to extract meaningful descriptions.
3. BLIP2 + GPT-4: Utilized GPT-4 to enhance the coherence and narrative flow of frame descriptions by integrating contextual information across consecutive frames. This approach helps mitigate inconsistencies in frame descriptions by leveraging temporal dependencies, ensuring that sequential descriptions maintain logical continuity. Additionally, GPT-4 can enrich the generated text with more descriptive language, filling in missing contextual details that might not be explicitly visible in a single frame.
4. BLIP2 + YOLO + GPT-4: Used YOLO for object detection and GPT-4 to refine descriptions, making them more detailed and structured. This approach aims to support accurate object recognition while transforming frame descriptions into sequences that may enhance contextual understanding.It is assumed that identifying key objects is essential for accurate action recognition, as it enhances the contextual relevance and specificity of frame descriptions.

Each configuration—baseline, edge detection, and contrast enhancement—will be evaluated as described in the 5.1.3 section. This study aims to quantify the contribution of these preprocessing techniques to the pipeline's performance and identify whether augmentations can effectively mitigate the challenges posed by low-quality video data.

## 5.2 Results

Table 1 showing Training Frame Descriptions vs. Real Frames. The cosine similarity results in Table 1 reveal that BLIP2 + GPT-4 consistently outperformed the baseline BLIP2, particularly in scenarios like Blowing Candles (0.1654 vs. 0.097) and Floor Gymnastics (0.1676 vs. 0.136). This suggests that GPT-4's textual refinement enhances the descriptive quality of training frame captions, leading to a better alignment with real frame descriptions. Conversely, BLIP2 + Edge Detection showed the weakest performance (average 0.0668), likely due to its focus on structural rather than contextual features, limiting its semantic richness.

Table 2 showing Generated Frame vs. Next Original Frame. The frame prediction results in Table 2 indicate that BLIP2 alone produced the highest cosine similarity scores across multiple test cases, reinforcing its advantage in direct visual recognition. BLIP2 + Edge Detection exhibited the lowest performance, with scores dropping significantly (e.g., 0.1723 in Blowing Candles compared to BLIP2's 0.2982). This suggests that while edge detection enhances structural boundaries, it fails to contribute meaningfully to predictive frame descriptions, possibly introducing unnecessary noise.

Adding GPT-4 improved results but did not surpass the baseline BLIP2 in all cases. This indicates that while GPT-4 enhances descriptions semantically, it may introduce speculative content that misaligns with actual frame progressions. YOLO's integration with GPT-4 improved object detection but did not consistently outperform standalone GPT-4 augmentation, likely due to over-reliance on bounding boxes restricting flexibility in complex motion sequences.

The statistical tests, summarized in Tables 3, 4, 5, 6, confirm that most model comparisons exhibited highly significant differences. For example, in Bowling, all p-values were 0 except for the BLIP2 + GPT-4 vs. BLIP2 + GPT-4 + YOLO comparison (p = 0.7565), indicating that YOLO's contribution to GPT-4's performance was minimal for this case. Similarly, in Clean and Jerk, the BLIP2 vs. BLIP2 + Edge Detection comparison had a p-value of 0.026, suggesting that their performance was relatively close. This highlights cases where Edge Detection, while often underperforming, provided comparable results in some structured activities. For Blowing Candles and Floor Gymnastics, all p-values were 0, indicating highly significant performance differences among models. This suggests that different augmentation techniques significantly impact predictive alignment depending on the nature of the action in the video. The t-tests confirm that BLIP2 consistently outperforms BLIP2 + Edge Detection across all tested actions. The consistently low p-values (<= 0.0001) in multiple comparisons suggest that Edge Detection does not enhance frame prediction accuracy and, in some cases, introduces noise that negatively affects performance. However, in Clean and Jerk, the p-value of 0.026 suggests that in certain structured scenarios, the effect of Edge Detection was not entirely detrimental, warranting further investigation. The BLIP2 + GPT-4 pipeline showed statistically significant differences from BLIP2 alone, particularly in Blowing Candles and Floor Gymnastics (p = 0.0000). This suggests that textual enhancements improve contextual richness in descriptions, though the added complexity occasionally leads to inconsistencies, as seen in Bowling, where p = 0.7565 indicated that the addition of GPT-4 did not significantly differ from GPT-4 + YOLO. The t-test comparisons reveal that adding YOLO to the BLIP2 + GPT-4 pipeline did not consistently enhance performance. For instance, in Bowling, the p-value of 0.7565 suggests that the difference between BLIP2 + GPT-4 and BLIP2 + GPT-4 + YOLO was not statistically significant. This indicates that while YOLO improves object recognition, its contribution to overall frame prediction accuracy remains limited, possibly due to its reliance on bounding boxes that restrict the model's ability to generalize across complex sequences. Given the results, an important observation is that BLIP2 alone remains the most reliable model for direct frame prediction, while GPT-4 provides beneficial but inconsistent improvements in descriptive accuracy. The integration of YOLO enhances object recognition but does not always translate to improved predictions,

**Table 1: Cosine Similarity: Training Frame Descriptions vs. Real Frames**

| *Topic/Model* | *BLIP2* | *BLIP2 + Edge Detection* | *BLIP2 + GPT-4* | *BLIP2 + GPT-4 + YOLO* |
|---|---|---|---|---|
| Blowing Candles | 0.097 | 0.0694 | **0.1654** | 0.1444 |
| Bowling | 0.0979 | 0.0555 | **0.1599** | 0.1587 |
| Clean and Jerk | 0.0594 | 0.0683 | 0.1129 | **0.1247** |
| Floor Gymnastics | 0.136 | 0.0741 | **0.1676** | 0.1464 |
| Average | 0.0975 | 0.0668 | **0.1514** | 0.1434 |

**Table 2: Cosine Similarity: Generated Frame vs. Next Original Frame**

| *Topic/Model* | *BLIP2* | *BLIP2 + Edge Detection* | *BLIP2 + GPT-4* | *BLIP2 + GPT-4 + YOLO* |
|---|---|---|---|---|
| Blowing Candles | **0.2982** | 0.1723 | 0.2656 | 0.2774 |
| Bowling | **0.3138** | 0.2265 | 0.3031 | 0.3026 |
| Clean and jerk | **0.2914** | 0.2119 | 0.2378 | 0.2266 |
| Floor Gymnastics | **0.3064** | 0.1462 | 0.2828 | 0.2723 |
| Average | **0.3021** | 0.1892 | 0.2723 | 0.2697 |

particularly when bounding box constraints limit adaptation to dynamic environments. Had the dataset contained more complex motion sequences or less structured activities, the reliance on GPT-4 and YOLO might have been more advantageous.

The highly significant p-values (p = 0.0000) in Blowing Candles and Floor Gymnastics indicate that different augmentation techniques have substantial impacts on frame prediction accuracy, depending on the complexity of the action. This suggests that future work should explore hybrid models that balance textual augmentation with direct visual recognition.

Overall, the t-test results reinforce the importance of statistical validation in model performance evaluation. The significant differences observed in multiple comparisons highlight the need for selecting augmentation techniques based on the specific characteristics of the video sequence rather than applying a one-size-fits-all approach.

## 5.3 Discussion

The performance evaluation of BLIP2 and its alternatives reveals key insights into the relationship between textual augmentation and frame prediction accuracy. BLIP2 demonstrates superior predictive capabilities by relying exclusively on direct visual recognition, avoiding the inconsistencies introduced by speculative textual augmentation. Conversely, Edge Detection, while useful for structural enhancement, lacks the semantic depth required for generating meaningful frame descriptions, leading to diminished performance. YOLO, when integrated, enhances object recognition in some cases, but its reliance on predefined bounding boxes restricts its adaptability in complex video sequences. These findings underscore the strengths and limitations of each approach and highlight the factors that influence predictive effectiveness.

Frame prediction results further indicate that multimodal approaches improve frame description precision, even though direct visual recognition remains the most effective method for achieving high prediction accuracy. The inconsistencies observed in certain models can be attributed to their reliance on textual inferences, which, while enhancing descriptive richness, occasionally introduce distortions that misalign with the expected frame progression. The observed deviations suggest that while textual augmentation aids in enriching contextual understanding, it does not always translate to improved predictive performance.

One critical limitation that emerges from the analysis is the role of dataset complexity. The UCF101 dataset, characterized by relatively simple and structured actions, provides a controlled testing environment. However, the effectiveness of LLM-based predictions may vary when applied to more dynamic or ambiguous sequences. Selecting a dataset with greater variability in motion, lighting, and environmental transitions would allow for a more comprehensive evaluation of these models' robustness. Such a dataset would provide a more rigorous test of their adaptability in challenging conditions, highlighting potential weaknesses and areas for improvement in predictive performance.

A key distinction must be drawn between training frame descriptions and next-frame predictions. Training frames provide static, observable details, whereas predictions inherently involve dynamic reasoning about future states. The results indicate that certain models tend to exaggerate motion expectations, predicting abrupt transitions where only minimal changes are present. This behavior is particularly evident in text-driven models, where inferred narrative structures may lead to overestimated action continuity.

Another critical aspect is context retention—the ability of a model to maintain logical continuity between frames. Analyzing mispredictions reveals that certain models struggle with object permanence, resulting in inconsistencies across sequential descriptions. This challenge is particularly pronounced in scenes featuring occlusions or environmental variations, which are common within the dataset. Addressing this issue may require refining temporal dependencies within the model architecture to ensure greater consistency in frame-to-frame transitions.

**Table 3: T-test: Bowling Example**

| **Bowling** | **BLIP2** | **BLIP2 + Edge Detection** | **BLIP2 + GPT-4** | **BLIP2 + GPT-4 + YOLO** |
|---|---|---|---|---|
| BLIP2 | | | | |
| BLIP2 + Edge Detection | 0.0000 | | | |
| BLIP2 + GPT-4 | 0.0000 | 0.0000 | | |
| BLIP2 + GPT-4 + YOLO | 0.0000 | 0.0000 | 0.7565 | |

**Table 4: T-test: Clean and Jerk Example**

| **Clean and Jerk** | **BLIP2** | **BLIP2 + Edge Detection** | **BLIP2 + GPT-4** | **BLIP2 + GPT-4 + YOLO** |
|---|---|---|---|---|
| BLIP2 | | | | |
| BLIP2 + Edge Detection | 0.0260 | | | |
| BLIP2 + GPT-4 | 0.0000 | 0.0000 | | |
| BLIP2 + GPT-4 + YOLO | 0.0000 | 0.0000 | 0.0006 | |

**Table 5: T-test: Floor Gymnastics Example**

| **Floor Gymnastics** | **BLIP2** | **BLIP2 + Edge Detection** | **BLIP2 + GPT-4** | **BLIP2 + GPT-4 + YOLO** |
|---|---|---|---|---|
| BLIP2 | | | | |
| BLIP2 + Edge Detection | 0.0000 | | | |
| BLIP2 + GPT-4 | 0.0121 | 0.0000 | | |
| BLIP2 + GPT-4 + YOLO | 0.0000 | 0.0000 | 0.0000 | |

**Table 6: T-test: Blowing Candles Example**

| **Blowing Candles** | **BLIP2** | **BLIP2 + Edge Detection** | **BLIP2 + GPT-4** | **BLIP2 + GPT-4 + YOLO** |
|---|---|---|---|---|
| BLIP2 | | | | |
| BLIP2 + Edge Detection | 0.0000 | | | |
| BLIP2 + GPT-4 | 0.0000 | 0.0000 | | |
| BLIP2 + GPT-4 + YOLO | 0.0000 | 0.0000 | 0.0000 | |

The integration of Large Language Models (LLMs) into video prediction represents a departure from conventional CNN-based architectures, emphasizing textual reasoning over direct visual analysis. This approach introduces both advantages and limitations. On one hand, LLMs excel at contextualizing frame descriptions, generating semantically rich narratives that enhance interpretability. However, their reliance on textual inference, rather than spatial continuity, introduces the risk of hallucinations, where predictions deviate from logical frame progressions.

A key observation from this study is that multimodal integration—combining textual and visual processing—may offer a more balanced approach. While direct visual models like BLIP2 achieve the highest accuracy in frame prediction, the contextual enhancements provided by LLM-based descriptions could be leveraged to improve narrative consistency in more complex scenarios. Furthermore, the computational efficiency of LLM-driven prediction must be considered. Traditional video prediction models rely on frame-by-frame spatial encoding, whereas LLMs operate on sequential text representations. This structural difference suggests that hybrid approaches could potentially reduce computational overhead while maintaining predictive reliability, particularly in applications requiring long-range temporal coherence.

The limitations of our research significantly influenced our findings, particularly in how different models handled frame descriptions and predictions. Due to computational and budgetary constraints, we were unable to fully utilize APIs for refining prompts across a larger portion of our dataset, which may have affected the consistency and generalizability of our results. Additionally, the simplicity of the actions in the UCF101 dataset likely played a role in shaping the observed performance of the models. Since straightforward actions tend to produce broad, less detailed embeddings, there was a noticeable discrepancy between the embeddings created from image frames and the more specific text-based embeddings generated by LLMs. This misalignment may have contributed to inconsistencies in prediction accuracy, as models relying on textual augmentation—such as GPT-4—sometimes introduced contextual details that were not present in the actual video. Moreover, models like BLIP2, which relied primarily on direct visual features, demonstrated stronger performance in frame prediction precisely because they avoided speculative textual reasoning. However, the poor performance of BLIP2 combined with Edge Detection suggests that structural enhancements alone do not necessarily translate to improved textual descriptions, emphasizing the need for deeper semantic processing rather than purely visual augmentations. The dataset's relatively structured and predictable action sequences also meant that our results may not fully extend to more complex, dynamic environments where occlusions, irregular movement patterns, and environmental variations could introduce additional challenges. These limitations highlight the necessity of expanding future research to more diverse datasets, incorporating more nuanced motion sequences, and exploring refined multimodal approaches that balance textual reasoning with direct visual inference.

These findings highlight the evolving role of LLMs in video analysis and underscore the necessity for further refinement in balancing textual augmentation with direct visual inference. Expanding the evaluation to diverse datasets and refining multimodal interactions will be essential steps in advancing the field of LLM-based video prediction.

## 6 CONCLUSIONS AND FUTURE WORK

This study aimed to assess the potential of leveraging Large Language Models (LLMs) for next-frame video prediction without requiring extensive model training. Our findings confirm that LLMs can indeed facilitate the generation of accurate frame predictions, offering a computationally efficient and accessible option for users. This approach is particularly beneficial for those who prefer not to engage in the costly and time-consuming process of training models from scratch, and offers a user-friendly alternative that takes advantage of capabilities of existing models.

Our research did have some limitations, the first one being the restricted scope of our testing due to budgetary constraints specific to our testing environment. These constraints limited our ability to fully utilize APIs for prompt refinement on larger amount of the data at hand. Moreover, the simplicity of the actions in our dataset's frames may have impacted the utility of our metrics, which involved using CLIP to create text and image embeddings for comparisons. A straightforward action in a frame often results in a broader, more generalized embedding, whereas the text descriptions generated for these frames tend to be more detailed, resulting in more specific embeddings. This discrepancy can lead to some misalignment between the model's output and the actual "ground truth." Consequently, this limitation implies that our current results might not be fully applicable to scenarios involving more complex actions. We leave future work to apply our methodology to a wider range of datasets and explore alternative metrics that could capture more detailed interactions within video sequences.

Ultimately, despite these challenges, the method developed through this study offers a strong basis for the advancement of video prediction using LLMs and for researchers looking for cost-effective solutions in video processing technologies.

## REFERENCES

[1] Simone Bianco, Luigi Celona, Marco Donzella, and Paolo Napoletano. 2023. Improving Image Captioning Descriptiveness by Ranking and LLM-based Fusion. arXiv:2306.11593 [cs.CV] https://arxiv.org/abs/2306.11593

[2] Padmashree Desai, C Sujatha, Saumyajit Chakraborty, Saurav Ansuman, Sanika Bhandari, and Sharan Kardiguddi. 2022. Next frame prediction using ConvLSTM. *Journal of Physics: Conference Series* 2161, 1 (jan 2022), 012024. https://doi.org/10.1088/1742-6596/2161/1/012024

[3] Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z Li. 2022. Simvp: Simpler yet better video prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3170–3180.

[4] Shreyank N Gowda, Marcus Rohrbach, and Laura Sevilla-Lara. 2021. Smart frame selection for action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 1451–1459.

[5] Xianfan Gu, Chuan Wen, Weirui Ye, Jiaming Song, and Yang Gao. 2024. Seer: Language Instructed Video Prediction with Latent Diffusion Models. arXiv:2303.14897 [cs.CV] https://arxiv.org/abs/2303.14897

[6] Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. 2024. VTimeLLM: Empower LLM to Grasp Video Moments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 14271–14280.

[7] Cheng Huang and Hongmei Wang. 2020. A Novel Key-Frames Selection Framework for Comprehensive Video Summarization. *IEEE Transactions on Circuits and Systems for Video Technology* 30, 2 (2020), 577–589. https://doi.org/10.1109/TCSVT.2019.2890899

[8] Shruti Jadon and Mahmood Jasim. 2020. Unsupervised video summarization framework using keyframe extraction and video skimming. In *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*. 140–145. https://doi.org/10.1109/ICCCA49541.2020.9250764

[9] Nishtha Jatana, Charu Gupta, Oday Hassen, and Ansam Abdulhussein. 2023. Future Frame Prediction using Generative Adversarial Networks. *Karbala International Journal of Modern Science* (12 2023).

[10] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*. PMLR, 12888–12900.

[11] Ruyang Liu, Chen Li, Haoran Tang, Yixiao Ge, Ying Shan, and Ge Li. 2025. St-llm: Large language models are effective temporal learners. In *European Conference on Computer Vision*. Springer, 1–18.

[12] Ruipu Luo, Ziwang Zhao, Min Yang, Junwei Dong, Da Li, Pengcheng Lu, Tao Wang, Linmei Hu, Minghui Qiu, and Zhongyu Wei. 2023. Valley: Video Assistant with Large Language model Enhanced abilitY. arXiv:2306.07207 [cs.CV] https://arxiv.org/abs/2306.07207

[13] R OpenAI et al. 2023. GPT-4 technical report. *ArXiv* 2303 (2023), 08774.

[14] J Redmon. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*.

[15] Xi Ye and Guillaume-Alexandre Bilodeau. 2023. Video prediction by efficient transformers. *Image and Vision Computing* 130 (2023), 104612. https://doi.org/10.1016/j.imavis.2022.104612