Video Reverse-Engineering in Frontier Foundation Models: Are We Getting There?

Jerry Han
jerryhan@princeton.edu

Arjun Menon
armenon@princeton.edu

Addison Wu
addisonwu@princeton.edu

Abstract

Short-form video content has proliferated more rapidly than ever across the entire internet due to its relative ease of creation, and accessibility and receptibility to a wide variety of audiences around the world. Unfortunately, the mass dissemination of such content has also attracted malicious actors to use video content as a prominent vector of harmful content, benign forms – i.e. misinformation, deepfakes – of which are difficult to be reliably detected by current AI-powered moderation systems. As an intermediate sub-problem in tackling this widespread societal issue, we explore the feasibility of whether or not AI models are capable of detecting malicious and suspicious edits in short form video content, with a starting focus on pinpointing and timestamping cut transitions. We explore the use of frontier foundational models, particularly Google's Gemini 1.5 Pro that has recently gained the ability to take in continual video input in addition to both text and still image, and compare its capabilities to an existing task-specific neural network TransNet V2. Our findings across a variety of quantitative and qualitative metrics shed light on the progress foundational models can create in regards to mass-scale short-form video moderation across the web, while also highlighting certain limitations.

1 Introduction and Related Work

The rising popularity of social media has dramatically transformed the way information is shared throughout the world, making such apps a critical vector for content propagation, given that they are so accessible by almost all members of society. In particular, short-form video platforms like Instagram and TikTok have become particularly potent channels for rapidly spreading information, but also increasingly sophisticated forms of misinformation [4]. While text-based content has developed relatively robust detection and flagging mechanisms [5], video content presents a more complex challenge, characterized by nuanced visual and contextual cues that make comprehensive monitoring significantly more difficult [6].

Many videos can be deemed harmful for public consumption based on visual cues that can clearly be observed at the surface level, for example, those that involve pornographic and gory content. While such visually explicit content can be relatively straightforwardly detected and removed through surface-level visual filters [7], a more insidious category of harmful videos has emerged that leverages sophisticated digital manipulation techniques. These covert forms of misinformation employ advanced technologies like deepfakes, which can fabricate entirely fictional scenarios featuring real people, creating highly convincing false narratives that can dramatically influence public perception. Strategic editing techniques, including selective clip placement, misleading audio overlays, and contextual manipulation, allow creators to fundamentally alter the meaning of visual content in ways that are challenging to immediately detect [3].

Recent advancements in artificial intelligence, particularly in multimodal large language models, offer promising avenues for addressing these challenges. Notably, Google's Gemini 1.5 Pro is one of the only ones that demonstrates remarkable capabilities in processing and understanding video-based content [2]. While addressing the overarching challenge of universal malicious edit detection in videos is not currently feasible to conduct in one work, we focus on a critical yet sanity-level computational task at the heart of this challenge: clip boundary detection. This work sits at the intersection of computer vision, natural language processing, and digital forensics, promising to provide crucial tools for maintaining information integrity in an increasingly complex digital ecosystem.

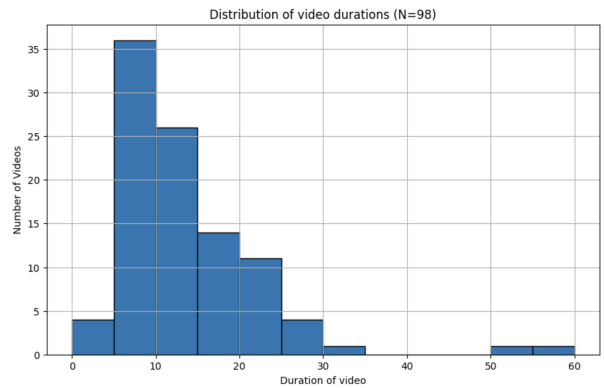
2 Methodology

2.1 Dataset

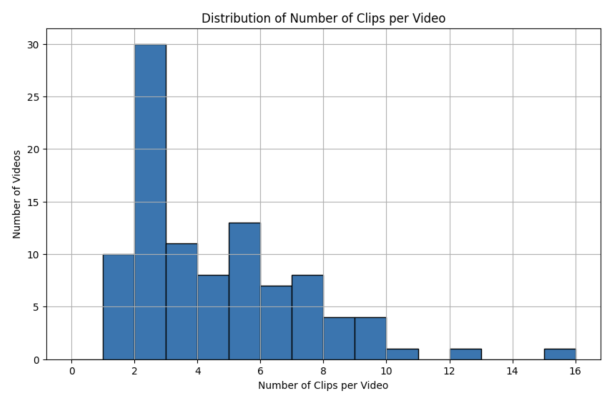
To accessibly construct a diverse dataset that would be well-representative of the short-form video content prevalent on today's social media platforms, we sourced videos posted by other users from the popular short-form video editing and design website CapCut [1]. Specifically, for evaluative purposes, we sourced $N = 98$ videos with a varying number of clips and total duration, all with a frame rate of 30 frames per second.

A script was developed which could scrape and download template videos from CapCut's template home page. However, the ground truth shot boundaries could only be obtained manually, accounting for the relatively small size of the dataset.

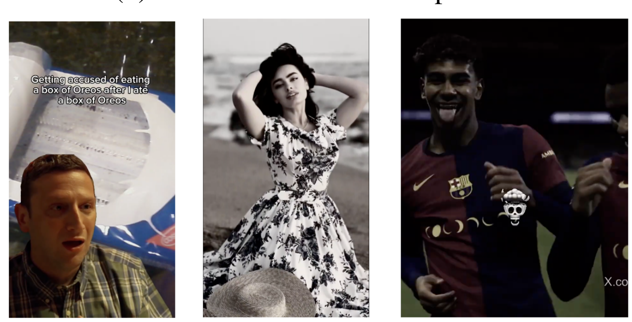
Further details regarding the distribution of our dataset, in terms of both duration and number of clips, are depicted in 4a and 4b, with example images from our dataset depicted in 4c. While the number of clips in each video is rather uniform (save for an outlier of 3 clips with 30 videos) with 10 videos for each clip count from 1 to 8, the duration of the videos heavily skews right, with most durations being in the range of 5 to 20 seconds.



(a) Distribution of video duration length



(b) Distribution of video clip count



(c) Example clips of videos from the CapCut dataset

Figure 1

2.2 Pre-Evaluation Fine-Tuning

Once we had our dataset, we split our 98 videos into two groups: fine-tuning and evaluation. Our fine-tuning dataset consisted of 20 videos, chosen at random from our larger dataset, and our evaluation dataset consisted of the remaining 78 videos. The videos in the fine-tuning dataset, once downloaded, were pre-processed into tensor form and grouped with information about the number of clips in the video as well as the clip durations.

These clips were used to fine-tine a traditional video-understanding neural network known as TransNet V2 [9], about which more information can be in the inference section of the paper. An important consideration we had in regards to TransNet V2 was the data it was trained on. In particular, TransNet V2 was trained on clip times in medium to longer-form videos on the order of several minutes long, thus raising a possible concern of the inability to deal with shorter form video in which clips occur at a higher frequency and whose occurrence may merely be perceived as noise.

To alleviate this concern, we conducted a fine-tuning approach. We first froze all weights of the original TransNetV2 model except those in the final 2, and then proceeded to train the model further using the fine-tuning dataset using ADAM as the optimizer. As the model was initially built to target medium to long-form data and we wanted to tune it for short-form data, we fine-tuned the model for 10 epochs to ensure it was able to learn to handle shorter, faster changes.

2.3 Inference

Each of the $N = 78$ evaluation videos were then sent as API queries to Gemini Flash 1.5 with the following prompt:

```
Analyze this video and determine the starting
and ending timestamps for each clip in the
video. For each clip provide a brief
description of the content in the clip.
Provide the output in JSON format with
the timestamps listed in the
Minutes:Second.millisecond format.
The output should be in this format, a
list of clips
[
{ "clip": ...,
"start": minutes:seconds.milliseconds,
"end": minutes:seconds.milliseconds },
...
]
```

A sample response from the API request is as follows:

```
[{'clip': 'The video starts with a shot
of a sunset in the distance from a car
driving down the highway.  Text over
the video reads "why do you always post
abt Jesus??" Then text reading "I
won\'t be quiet!! my God is alive!!
So how could I keep it inside?"  The
video then cuts to a shot of a clear
blue sky with clouds moving in the
distance.',
'start': '0:00.000',
'end': '0:07.000'},
{'clip': 'Text reading "PRAISE
the Lord! oh my soul!!"',
'start': '0:07.000',
'end': '0:13.000'}]
```

2.3.1 TransNet V2

To better understand the relative capabilities of Gemini 1.5 Flash in inverse video edit detection, inference was also run using the state-of-the-art TransNet V2 neural network created for shot boundary detection [9]. The TransNet V2 architecture incorporates a sub-structure called a *dilated deep convolutional neural network* (DDCNN) that is made up of multiple 3 x 1 x 1 convolutional layers followed by another layer of the same dimension, in parallel (four streams), with all outputs then being concatenated and batch-normalized.

After the images are pre-processed in the

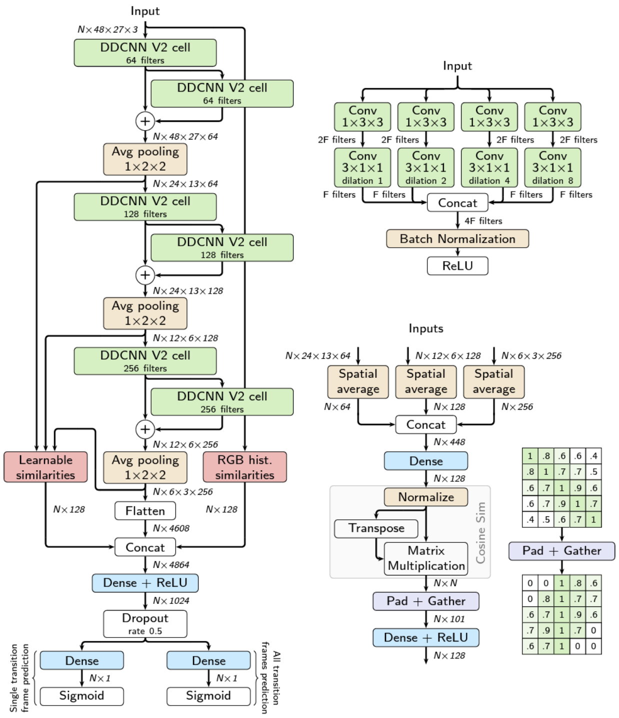


Figure 2: TransNet V2 (left), Dilated Deep Convolutional Neural Network Cell Substructure (upper right), Learnable Similarity Mechanism (lower right) [9]

DDCNN substructures, the outputs are then fed into learnable similarity blocks in which matrix multiplications aim to capture temporal similarity relationships between nearby images in the continual video sequence, with the main motivation being that close image pairs of lower similarity metrics correspond to a cut transition. The resultant outputs of the network are positive and negative labels for each frame in the video as to whether or not they are part of a start-end frame pairing representing a cut transition.

3 Evaluation

As a sanity check, we first evaluated the models using the following metrics

1. Duration deviation: Average percentage deviation in predicted total clip duration in a video

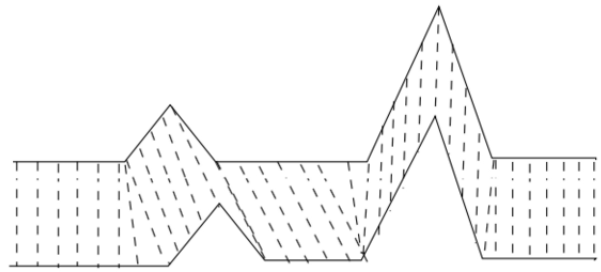


Figure 3: Visualization of the dynamic time warping (DTW) algorithm, with dashed lines representing DTW alignment between pairs of timestamps

2. Clip segmentation accuracy: Number of videos with number of clips predicted correctly divided by total number of videos

One primary concern we had with the basic metric of frame classification accuracy used in [9], as in whether or not each frame was correctly classified as being associated with a cut transition, was its brittleness in regards to edge cases or situations where the classification might be technically correct but fails to capture the broader context or intent of the transition. For instance, a model might correctly identify isolated cut transitions while missing the overall temporal structure or misclassifying transitions in complex sequences, leading to misleadingly high accuracy metrics that do not reflect real-world performance.

Thus, we use a sophisticated metric to evaluate the similarity ground truth vs predicted clip boundaries by treating these as temporal sequences, and then applying the dynamic time warping (DTW) algorithm [8]. DTW is an algorithm that aligns two sequences by warping their time axes non-linearly to find an optimal match between them. Unlike simple distance measures like Euclidean distance, DTW can handle sequences that are similar but out of phase in the time dimension. DTW aligns both temporal sequences in a way that minimizes the total distance between them, and then reports this distance as the DTW loss. An example visualization of this algorithm is shown in 3.

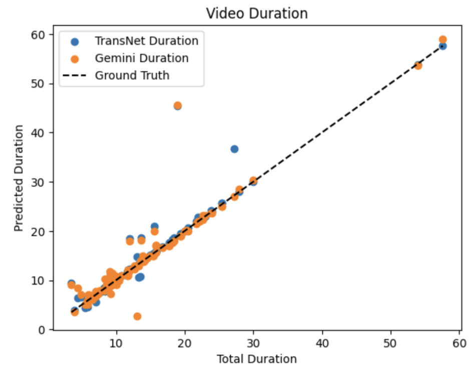
Table 1: Comparison of Gemini and TransNet Across Metrics

Metric	Gemini	TransNet
Clip Segmentation Accuracy	48.0%	39.8%
Duration Deviation	7.56%	10.61%
Mean DTW Loss	14.5	30.2

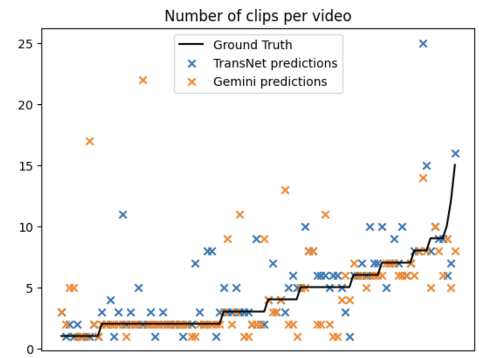
We deduce that Gemini Flash 1.5 outperforms TransNet V2 in terms of shot boundary detection accuracy, as evidenced by the higher clip segmentation accuracy, lower duration deviation and lower mean DTW loss (average DTW loss over all videos in the dataset).

Examining the predicted total video durations, we noticed that both models seemed to perform quite well, getting very close to the correct duration but with small quantities of deviation. The total duration estimates were calculated by summing up all the estimated clip durations produced by both models, with any deviation from the true values being caused by the model detecting "overlapping" clips, or the same portion of one clip as being part of two different clips. This hiccup seemed to occur more for TransNet than Gemini, and the average deviation was lower for Gemini due to this. Notably though, when Gemini did hiccup, it could be quite seriously. As visible in 4a, Gemini has 2 significant outliers that are visibly quite far away from the equilibrium line, in which it predicted a 12-second video to be around 1-second long and a 20-second video to be around 45-seconds long. TransNet only produced one such outlier. As Gemini analyzes the video as a whole as a opposed to TransNet's systematic frame-by-frame method, it may be more prone to making large errors of this kind, with a small issue within the video causing cascading effects on its predictions.

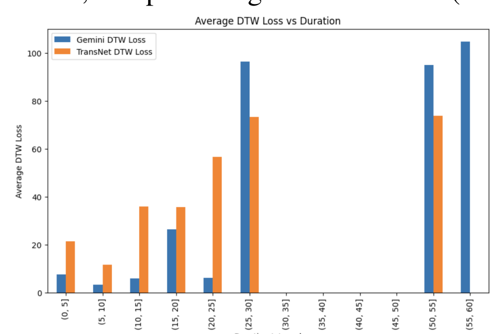
The predicted clip counts seemed to follow



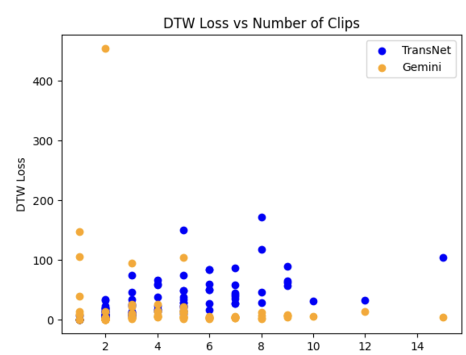
(a) Predicted video durations generated by Gemini and TransNet V2, compared to ground truth line of $y = x$



(b) Predicted clip counts generated by Gemini and TransNet V2, compared to ground truth line (black)



(c) Average DTW loss with respect to video length



(d) DTW loss with respect to video clip count

Figure 4

similar patterns to the predicted video durations. Gemini performed better than TransNet overall, both by pure accuracy and average deviations. However, most of TransNet's deviations were small, as it was found to often predict either 1 additional clip, or 1 fewer clip than ground truth. We ascribe this to TransNet picking up video artifacts (potentially introduced by editing bugs), which Gemini is able to avoid due to it's understanding of the visual content in the video. However, when Gemini did make an error, it tended to be a significant one. Gemini had around 4 significant outliers where its predicted clip count was incredibly far from the true value, with especially notable cases where it predicted 17 and 22 clips for videos that had 2 clips in reality. These clips may have been especially noisy, containing subjects that moved around in a hard to follow manner, prompting Gemini to categorize their behavior within the clips as a transition or clip change. Further qualitative analysis reveals Gemini performing worse on videos with "subtler" and more rapidly successive transitions as well, as can be seen in 5.

As expected, DTW loss increases with video duration but remains relatively constant as a function of the number of clips. Gemini's performance degrades much faster for videos exceeding 25s, compared to the decrease in TransNet's performance. This demonstrates Gemini's limited ability to effectively process and retain information for longer sequences, suggesting that despite its supposedly expanded long-context window, there are still challenges dealing with long-form video.

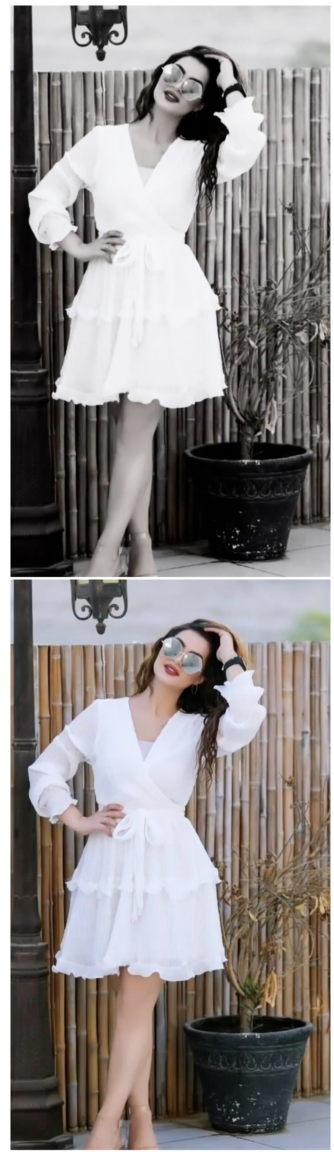


Figure 5: Similar 'before' (top) and 'after' (bottom) video frames between a cut transition, missed by Gemini 1.5

4 Analysis

In general, we conclude that

1. Frontier multimodal foundation models such as Gemini 1.5 are capable of identifying shot detection boundaries with accuracy on par with and even exceeding that of state-of-the-art deep convolutional neural network archi-

tectures (TransNet V2), especially for shorter-form video content. That being said...

2. Gemini 1.5 exhibits significantly worse deterioration in performance for longer duration videos compared with TransNet V2, which is a key limitation of current foundational models.

3. We successfully achieved our goal of answering whether frontier foundation models are capable of video reverse-engineering, with our answer being: Yes, these models are able to identify shot transitions accurately, but only for short-form videos of duration <25s.

We acknowledge that there are some limitations with our current study, such as the limited size of our dataset, due to time limitations and the lack of an automated scraping mechanism to scrape ground truth labels from CapCut. A script to scrape video links from CapCut was developed, and further work could involve developing methods to scrape shot transition effects and timestamps directly from the CapCut editor interface.

The metrics used in our evaluation are rather rudimentary as well; more sophisticated metrics such as MeanIoU can yield more in-depth insights by taking into account both false positives and false negatives. This would involve simply adapting the metric (which was intended for evaluating image segmentations) to the temporal domain.

We also chose these more rudimentary metrics largely because of the limited size of our dataset; curating a larger dataset would allow us to segment the dataset further and perform a more extensive analysis in terms of the following: type of video transition, distribution of cuts, content of video, etc.

Further areas of improvement include developing ensemble models, where videos can be first passed through an established shot-detection model (such as TransNet V2); and then passed through our foundation model (with the TransNet outputs concatenated). This might yield an even higher accuracy and bring us closer to the goal of full video reverse engineering, and serves as a stepping stone to the ultimate goal of unsupervised video editing models.

References

[1] Capcut.

[2] Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context, 2024.

[3] ABDULLAH, S. M., CHERUVU, A., KANCHI, S., CHUNG, T., GAO, P., JADLIWALA, M., AND VISWANATH, B. An analysis of recent advances in deepfake image detection in an evolving threat landscape. In *2024 IEEE Symposium on Security and Privacy (SP)* (2024), pp. 91–109.

[4] BU, Y., SHENG, Q., CAO, J., QI, P., WANG, D., AND LI, J. Combating online misinformation videos: Characterization, detection, and future directions. In *Proceedings of the 31st ACM International Conference on Multimedia* (2023), Association for Computing Machinery, p. 8770–8780.

[5] KOU, Y., AND GUI, X. Flag and flaggability in automated moderation: The case of reporting toxic behavior in an online game community. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2021), CHI '21, Association for Computing Machinery.

[6] MA, R., AND KOU, Y. "how advertiser-friendly is my video?": Youtuber's socioeconomic interactions with algorithmic content moderation. *Proc. ACM Hum.-Comput. Interact. 5*, CSCW2 (Oct. 2021).

[7] MEJHED MKHININI, M., SIDIBE, A. S., BENALI, K., BENTAARIT, N., AND KHELIFI, A. Image and signal processing to detect violent content in social media videos. In *Proceedings of the 2023 15th International Conference on Machine Learning and Computing* (New York, NY, USA, 2023), ICMLC '23, Association for Computing Machinery, p. 309–315.

[8] SAKOE, H., AND CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing 26*, 1 (1978), 43–49.

[9] SOUCEK, T., AND LOKOC, J. Transnet v2: An effective deep network architecture for fast shot transition detection. In *Proceedings of the 32nd ACM International Conference on Multimedia* (New York, NY, USA,

2024), MM ’24, Association for Computing Machinery, p. 11218–11221.