A. Implementation details

All evaluations could be conducted on a single Nvidia A100 80G graphics card. To accelerate inference, we use a Linux server equipped with 8 Nvidia A100 80G cards. We carry out our evaluation across three model series and five model size. The weights for these models are available on Huggingface[1][2][3]. We implement rotary position embedding (RoPE) and apply a scaling factor of 2, extending the original context length from 4096 to 8192 tokens.

B. Time Consumption Experiment

we conducted the experiments using the same hardware specifications. Table 5 below shows the time consumption for inference with 500 samples from EgoSchema using a single NVIDIA A100 GPU. We conduct another detailed comparison in Table 6 with different numbers of input frames.

	SlowFast-LLaVA	**DyTo**
Dataset	Egoschema	
Model	LLaVA-NeXT-34B	
Input	$50 + 10$ frames	100 frames
Merge Strategy	Pooling	Dynamic token merging
Device	1 Nvidia A100 GPU	
Time Consumption	5.74 s/item	6.22 s/item

Table 5. Computational cost comparison with the baseline.

Input Frame Length	**Selected Frame(avg.)**	**Cluster time(ms)**	**Merge time(ms)**	**Token Count(avg.)**	**Egoschema** acc. (%)	**NeXTQA** acc. (%)	**IntentQA** acc. (%)	**VideoMME** acc. (%)	**MVBench** acc. (%)
100	9.0	430	5	2618	**48.6**	**65.7**	**61.6**	41.2	**45.2**
200	17.5	870	15	6390	47.0	65.0	59.7	**41.9**	43.3
300	24.3	1320	21	10315	45.2	63.7	58.0	40.6	41.2

Table 6. Computational cost comparison with different input frames.

As shown in the tables, the difference in time consumption is negligible. Although our method is slightly slower than SlowFast, we think it may be attributed to hardware optimizations or variance.

C. Performance on video fine-tuned model.

To demonstrate our model could work on all LVLMs, including video-finetuned model, we conduct experiments on LLaVA-NeXT-Video-7B and InternVL2-8B across various video understanding benchmarks. The results show that DyTo leads to improvements in nearly all the tasks.

Method	**Base Model**	**Input Frame Length**	**Token Length**	**NExTQA** acc. (%)	**EgoSchema** acc. (%)	**IntentQA** acc. (%)	**VideoMME** acc. (%)	**MVBench** acc. (%)
Official	LLaVA-NeXT-Video	16	2304	64.3	45.2	61.9	41.5	44.2
DyTo	LLaVA-NeXT-Video	100	2304	**65.5**	45.2	**62.0**	**41.8**	**44.8**
Official	InternVL2	16	4096	80.9	66.4	81.8	53.3	65.5
DyTo	InternVL2	100	4096	**81.4**	**67.6**	**83.0**	**59.3**	**66.2**

Table 7. DyTo Performance on different video fine-tuned models.

[1] https://huggingface.co/collections/liuhaotian/llava-16-65b9e40155f60fd046a5ccf2
[2] https://huggingface.co/OpenGVLab/InternVL2
[3] https://huggingface.co/Qwen/Qwen-VL-Chat

D. Sampling effectiveness of DyTo

Fig. 7 shows a visualization of 8 keyframes sampled from both DyTo and SlowFast. We can see that DyTo successfully extracted the key frames, which capture the entire event of washing clothes (highlighted in the red box).

Figure 7. Sampling Results.

E. Visualizations of Dynamic Bipartite Merging

To help understand dynamic token merging effectively, we provide a visualization comparing our method with the pooling method. As shown in Fig. 8, the proposed approach effectively maintains the object's actions while making every effort to prevent the disruption of the original spatial information. We set the constant merge ratio of r=288 to enable a convenient and fair comparison with the pooling method, while r is a dynamic integer that varies based on the number of clusters in DyTo. It is important to emphasize that our proposed token merging method operates without the need for any labels. To create the visualizations in Figure 8 , we follow each final merged token back to its original input patches. For each token, we color its corresponding input patches, referred to as "Patchified," using the average color of that region. To ensure that different tokens are distinguishable, we assign each token a random border color. It's important to note that tokens do not necessarily correspond to contiguous input regions. The only spatial information comes from the position encodings.

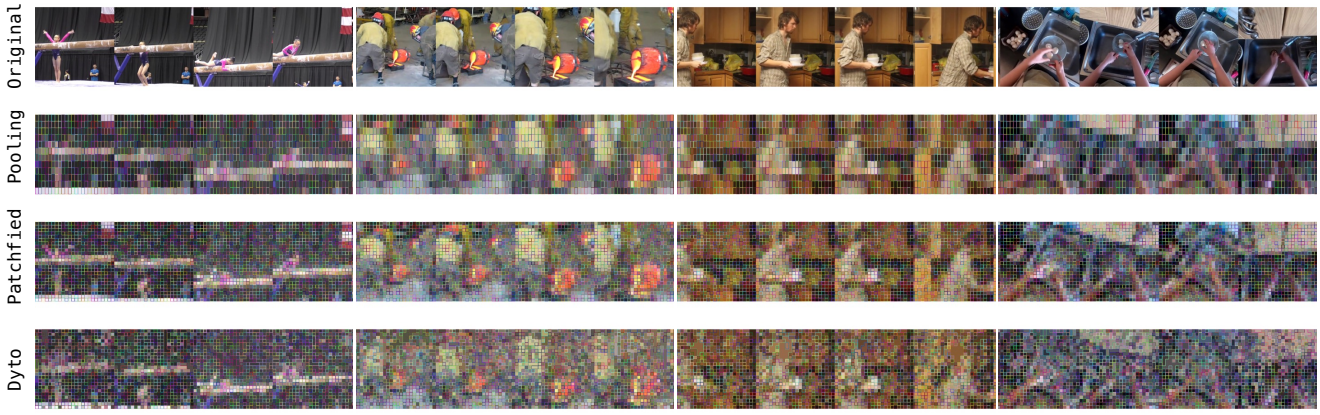
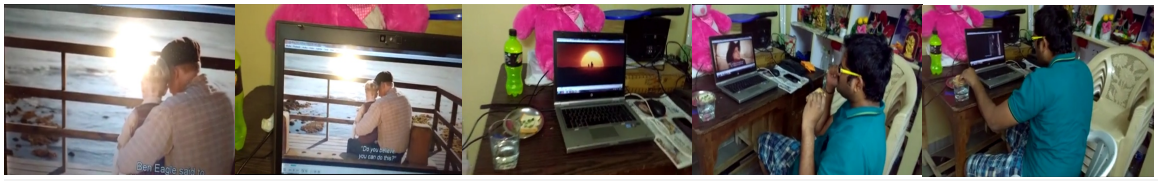


Figure 8. Results of merging on video frames.From top to bottom is original images,after pooling, patchfied, and the proposed bipartite merging. Patchfied means

F. More Qualitative Cases



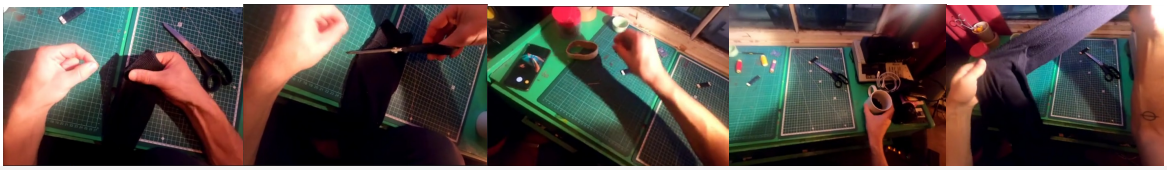
Question: Describe what happens in this video in detail.

IG-VLM-7B: A man is seen sitting in front of a laptop, watching a movie. The laptop is placed on a wooden desk. The man is wearing glasses and a blue shirt. The movie he is watching appears to be a romantic drama, as suggested by the content displayed on the laptop screen. The overall context of the video seems to be a casual, indoor setting, possibly a home or office environment, where the man is enjoying some leisure time by watching a movie.

SlowFast-LLaVA-7B: In this video, a man is sitting at a desk with a laptop in front of him. He appears to be engaged in a video call with another person, who is not visible in the image. The man is wearing a green shirt and plaid pants. On the desk, there is a glass of water and a sandwich. The setting suggests a casual, indoor environment, possibly a home or office. The man seems to be in a relaxed posture, indicating a comfortable and informal interaction. The context of the video suggests a personal or professional video call, possibly for work or personal reasons.

DyTo-7B: In this video, a man is seen sitting at a desk with a laptop, engaging in an activity that involves a teddy bear. The scene appears to be set in a room with a window, and there are other objects on the desk, such as a cup and a bottle. The man seems to be focused on the laptop screen, possibly working or browsing the internet. The presence of the teddy bear suggests a personal or playful element to the scene. The overall content of the video seems to be a casual, everyday moment captured in a home setting.

Figure 9. Example from MVBench benchmark. red part denotes the incorrect or hallucination content. green part denotes the correct object content.



Question: Describe what happens in this video in detail.

IG-VLM-7B: A person is shown sewing a garment, specifically a pair of black pants. The sequence of images captures the process of sewing, from the initial preparation of the fabric and thread, to the actual stitching, and finally the completion of the garment. The person is using a sewing machine and various sewing tools, such as scissors and a needle. The video provides a detailed look at the craftsmanship involved in creating clothing.

SlowFast-LLaVA-7B: A person is seen sewing a garment, specifically a pair of pants. The sequence of key frames captures the process of sewing, starting with the person cutting the fabric, then stitching the pieces together, and finally adjusting the fit of the pants. The video provides a detailed look at the craftsmanship involved in creating clothing, showcasing the skill and precision required in sewing.

DyTo-7B: A person is shown in the process of sewing a garment. The sequence of key frames captures the following events:
1. The person is sitting at a table with fabric.
2. They are holding a piece of fabric in their hands, preparing to sew it.
3. The person is using a sewing machine, which is visible in the image.
4. They are sewing the fabric onto another piece of fabric, which is also visible.
5. The person is holding a pair of scissors, likely for cutting the fabric.
6. The person is using a measuring tape, which is also visible in the image.
7. The person is holding a cup, possibly containing a beverage.
8. The person is wearing a tattoo on their arm.

Throughout the video, the focus is on the hands-on process of sewing, with the person demonstrating various tools and techniques. The setting appears to be a workspace or a crafting area, with the person engaging in a creative activity.

Figure 10. Example from Egoschema benchmark. green part denotes the correct object content. blue part denotes the missing content in IG-VLM and SlowFast-LLaVA answers. DyTo can effectively capture the events in the video.