When Spatial meets Temporal in Action Recognition

Huilin Chen[1,*] Lei Wang[1,2,*,†] Yifan Chen[1,*] Tom Gedeon[3] Piotr Koniusz[2,1]
[1]Australian National University, [2]Data61/CSIRO, [3]Curtin University
{u7326198, lei.w, u7725118}@anu.edu.au,
tom.gedeon@curtin.edu.au, piotr.koniusz@data61.csiro.au

Abstract

*Video action recognition has made significant strides, but challenges remain in effectively using both spatial and temporal information. While existing methods often focus on either spatial features (*e.g.*, object appearance) or temporal dynamics (*e.g.*, motion), they rarely address the need for a comprehensive integration of both. Capturing the rich temporal evolution of video frames, while preserving their spatial details, is crucial for improving accuracy. In this paper, we introduce the Temporal Integration and Motion Enhancement (TIME) layer, a novel preprocessing technique designed to incorporate temporal information. The TIME layer generates new video frames by rearranging the original sequence, preserving temporal order while embedding N^2 temporally evolving frames into a single spatial grid of size $N \times N$. This transformation creates new frames that balance both spatial and temporal information, making them compatible with existing video models. When $N = 1$, the layer captures rich spatial details, similar to existing methods. As N increases ($N \geq 2$), temporal information becomes more prominent, while the spatial information decreases to ensure compatibility with model inputs. We demonstrate the effectiveness of the TIME layer by integrating it into popular action recognition models, such as ResNet-50, Vision Transformer, and Video Masked Autoencoders, for both RGB and depth video data. Our experiments show that the TIME layer enhances recognition accuracy, offering valuable insights for video processing tasks.*

1. Introduction

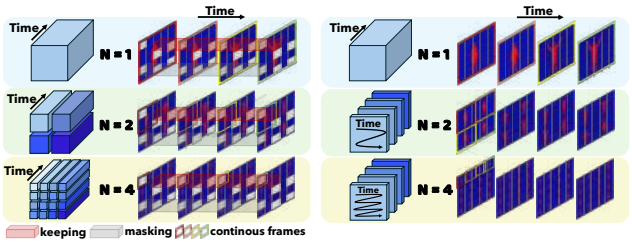
Video action recognition [2, 4, 5, 10, 12, 21, 24, 26, 28–33, 35, 36, 38, 41] has become a pivotal area in computer vision, with applications spanning from autonomous vehicles and surveillance systems to human-computer interaction and sports analytics. Despite the progress made by advanced 3D convolutional networks [4, 29] and transformer-based models [2, 8, 21], a core challenge remains: efficiently capturing both visual appearances and temporal dynamics within videos without excessive computational cost. Many models rely on sparse frame sampling [32] to limit memory usage, potentially overlooking critical motion that is essential for understanding complex actions [5, 21].

Recent research has emphasized the need for approaches that can flexibly adapt to both spatial and temporal demands of video data [5, 37]. However, methods that heavily rely on densely sampled sequences often encounter scalability issues, while those focused on spatial detail alone may underperform in capturing motion cues [28, 35]. To bridge this gap, we introduce the Temporal Integration and Motion Enhancement (TIME) layer, a preprocessing layer designed to maximize the temporal richness in video frames while preserving spatial fidelity, transforming standard image classifiers like ResNet [13] and Vision Transformer (ViT) [8] into efficient video action recognition models. By enhancing input frames with motion cues, the TIME layer addresses the limitations of sparse sampling approaches and enables detailed, continuous temporal analysis within each frame, even for models traditionally optimized for static images.

The TIME layer's adaptable design offers several advantages: (i) It balances spatial and temporal information through an adjustable grid structure, allowing possible tuning based on the specific action recognition requirements. (ii) The TIME layer operates independently of the model architecture, making it compatible with a wide range of models, from CNNs to transformers, as well as with emerging self-supervised learning approaches like VideoMAE. (iii) It facilitates systematic evaluation of spatial-temporal information by arranging frames in novel configurations, providing a new avenue for investigating how model architectures absorb and interpret motion and visual cues. This versatility enables TIME to serve as both an effective video preprocessing tool and a diagnostic layer, offering insights into model responses to varied spatial-temporal compositions.

In this work, we evaluate the TIME layer on both RGB

*These authors contributed equally.
†Corresponding author.



(a) Spatial block arrangement. (b) Temporal block arrangement.

Figure 1. Variants of the TIME layer. The spatial block arrangement captures long-term motion spatially by incorporating broad visual changes across frames, while it emphasizes short-term motion within each frame's temporal sequence (tube masking, as shown in (a), temporally obscures short-term motion details for VideoMAE). In contrast, the temporal block arrangement records short-term motion frame by frame, yet captures long-term motion temporally across the sequence (tube masking here obscures long-term motion). Each approach provides a unique perspective on balancing spatial and temporal information through the spatial-temporal balance parameter, N, in video data.

and depth video datasets across different action recognition benchmarks, examining its impact on small-scale and large-scale datasets, from-scratch training versus fine-tuning, and CNN versus transformer. We also explore how TIME enhances temporal context within each frame, making it a valuable tool for applications requiring a nuanced balance between visual appearance and motion dynamics. Through experiments, we demonstrate that the TIME layer improves model performance and provides a systematic framework for analyzing the role of temporal information in action recognition. Our **main contributions** are as follows:

i. We introduce the TIME layer, a novel preprocessing approach that enhances temporal representation in video frames while preserving spatial fidelity.
ii. We demonstrate that the TIME layer is compatible with a broad range of models, including CNN, transformer, and self-supervised learning framework, making it a versatile component for existing pipelines.
iii. Extensive experiments show that the TIME layer improves performance across diverse action recognition benchmarks, including adapting RGB-pretrained models to depth videos.
iv. The TIME layer functions as a tool for systematically evaluating spatial-temporal information processing, offering new insights into model behavior on video data.

2. Related Work

In this section, we identify gaps in existing video action recognition methods and highlight the need for adaptable, computationally efficient layers like the TIME layer, which enrich temporal dynamics in standard architectures.

Video action recognition has progressed from early 2D CNNs that process frames individually [13] to more advanced 3D CNNs [4] and transformer-based models capable of capturing spatial-temporal dependencies [21]. Initial approaches, such as C3D [29], used 3D convolutions to model relationships across frames. Further improvements, like I3D [4] and SlowFast [11], introduced dual-pathway architectures to handle both slow and fast motion patterns, but with high computational costs. Recent transformers, including ViViT [1] and TimeSformer [2], use self-attention for long-term temporal modeling but are similarly resource-intensive, underscoring the need for efficient temporal processing. To capture motion dynamics, Taylor videos [37] and motion prompts [5] have been introduced to enhance temporal cues in video frames by emphasizing motion and object displacement.

Our TIME layer addresses these challenges by offering an architecture-agnostic, flexible module that enhances temporal information within frames, aiming to bridge gaps in computational efficiency and adaptability across models.

Sparse sampling and temporal efficiency. To mitigate the computational demands of video action recognition, sparse sampling techniques reduce the number of processed frames, lowering memory and processing costs. Temporal Segment Networks (TSN) [32, 34] popularized uniformly sampling frames to capture temporal moments without exhaustive frame processing. However, sparse sampling may miss fine-grained motion cues in longer sequences [15, 31]. Recent approaches, such as Temporal Pyramid Networks (TPN) [40] and Temporal Shift Module (TSM) [18], aim to improve temporal representation without added computational cost, although they still rely on selected frames.

In contrast, our TIME layer captures the entire video sequence by restructuring frames to embed richer temporal dynamics while preserving spatial details, achieving a more comprehensive motion representation.

Self-supervised video representation learning. Self-supervised learning (SSL) has gained traction in video modeling by enabling pretraining without labeled data. Early SSL methods [30, 33] used techniques like Bag of Words and Fisher Vectors to improve action recognition. Recent SSL models, VideoMAE [28] and its advanced version [35], use masked autoencoding to learn spatial-temporal features by reconstructing occluded regions, effectively capturing motion and visual cues. However, SSL methods typically depend on sparse frame sampling or masking on consecutive frames, which may limit their capacity to fully capture both short-term and long-term temporal patterns.

Our TIME layer reorganizes video frames, allowing an adjustable spatial-temporal balance and facilitating a deeper exploration of action recognition trade-offs across varying temporal scales.

Spatial-temporal balancing is essential for video action recognition. Many traditional approaches face challenges

in maintaining this balance, as more temporal information may reduce spatial clarity [37]. Methods like dynamic images [3], Temporal Context Networks [6], and X3D [9] capture multi-scale temporal features but often remain architecture-specific. Additionally, combining consecutive RGB channels into a single frame for better motion detection [14] has shown potential, as have Taylor videos for capturing dominant motion in actions [37]; however, these methods often lack the flexibility to record either finer temporal or spatial details.

Our TIME layer provides a systematic approach to balancing spatial and temporal information, positioning it as an adaptable tool for examining model sensitivities to spatial and temporal variations, thereby opening new avenues for video-related research, *e.g.*, action recognition.

3. Approach

In this section, we introduce our Temporal Integration and Motion Enhancement (TIME) layer and provide insights into our approach. We begin by defining the notations.

Notations. Let $\mathcal{I}_T = \{1, 2, \ldots, T\}$ represent an index set. Scalars are denoted by regular fonts, *e.g.*, x; vectors by lowercase boldface, *e.g.*, $\boldsymbol{x}$; matrices by uppercase boldface, *e.g.*, $\boldsymbol{X}$; and tensors by calligraphic letters, *e.g.*, $\mathbf{X}$.

3.1. TIME Layer

Consider a video represented by a sequence $\mathcal{F} = [\mathbf{F}_1, \mathbf{F}_2, \ldots, \mathbf{F}_T]$, where T is the total number of frames, and each frame $\mathbf{F}_t \in \mathbb{R}^{H \times W \times 3}$ ($t \in \mathcal{I}_T$). The TIME layer processes frames in temporal order, efficiently using all frames to preserve fine motion cues in contrast to traditional sparse sampling techniques [32]. The TIME layer is independent of the model's architecture, allowing flexibility in frame resolution and input count. Given a target resolution of $H^* \times W^*$ and a frame count of T^*, the layer arranges each new frame by organizing an $N \times N$ grid of temporally ordered placeholders. If the video length T is shorter than T^*N^2, frames are repeated to reach the necessary length; otherwise, T^*N^2 frames are sampled to maintain broad temporal coverage. To construct each new frame, we use two main approaches, each enhancing temporal detail in distinct ways. Fig. 1 shows the two variants of TIME layer. These methods are described below.

Spatial block arrangement. In this approach, we divide the T^*N^2 sequence into N^2 temporal blocks, each containing T^* consecutive frames:

$$\mathcal{T}_i = [\mathbf{F}_{(i-1)T^*+1}, \mathbf{F}_{(i-1)T^*+2}, \ldots, \mathbf{F}_{iT^*}], \tag{1}$$

where $i = 1, 2, \ldots, N^2$, and each $\mathcal{T}_i \in \mathbb{R}^{H \times W \times 3 \times T^*}$ represents a segment of T^* frames from the video. We then arrange these blocks spatially in an $N \times N$ grid to form a new frame $\mathcal{F}^{\text{spatial}}$, with each grid cell corresponding to frames from a specific segment:

$$\mathcal{F}^{\text{spatial}} = \begin{bmatrix} \mathcal{T}_1 & \mathcal{T}_2 & \cdots & \mathcal{T}_N \\ \mathcal{T}_{N+1} & \mathcal{T}_{N+2} & \cdots & \mathcal{T}_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{T}_{N(N-1)+1} & \mathcal{T}_{N(N-1)+2} & \cdots & \mathcal{T}_{N^2} \end{bmatrix}. \tag{2}$$

This spatial configuration in $\mathcal{F}^{\text{spatial}}$ captures the progression of T^* frames within each grid cell, offering a temporally enriched view of motion continuity.

Temporal block arrangement. Alternatively, we organize frames by dividing the T^*N^2 sequence into T^* blocks, each containing N^2 frames, enabling detailed temporal evolution within each new frame. The blocks are defined as:

$$\mathcal{S}_j = [\mathbf{F}_{(j-1)N^2+1}, \mathbf{F}_{(j-1)N^2+2}, \ldots, \mathbf{F}_{jN^2}], \tag{3}$$

where $j = 1, 2, \ldots, T^*$, and each $\mathcal{S}_j$ represents a contiguous segment of N^2 frames. For each block $\mathcal{S}_j$, we arrange the frames into an $N \times N$ grid to form a new frame $\mathbf{F}_j^{\text{new}}$:

$$\mathbf{F}_j^{\text{new}} = \begin{bmatrix} \mathbf{F}_{(j-1)N^2+1} & \cdots & \mathbf{F}_{(j-1)N^2+N} \\ \mathbf{F}_{(j-1)N^2+N+1} & \cdots & \mathbf{F}_{(j-1)N^2+2N} \\ \vdots & \ddots & \vdots \\ \mathbf{F}_{(j-1)N^2+(N-1)N+1} & \cdots & \mathbf{F}_{jN^2} \end{bmatrix}. \tag{4}$$

This results in each new frame $\mathbf{F}_j^{\text{new}}$ encapsulating temporally rich content within a spatial layout. For simplicity, we denote this operation as $\mathbf{F}_j^{\text{new}} = r(\mathcal{S}_j)$. Thus, we construct the sequence:

$$\mathcal{F}^{\text{temporal}} = [r(\mathcal{S}_1), r(\mathcal{S}_2), \ldots, r(\mathcal{S}_{T^*})], \tag{5}$$

which captures the temporal evolution across consecutive frames. Once the new video frames are structured, we resize them to match the model's input dimensions ($H^* \times W^*$, generally with $H^* = W^*$). Initially, data augmentations such as cropping, scaling, flipping, and rotation are applied to the original high-resolution frames, which are then resized to fit within the $N \times N$ grid, optimizing memory efficiency.

The TIME layer's flexibility is governed by the parameter N, balancing spatial and temporal detail. We refer to N as the spatial-temporal balance parameter. When $N = 1$, the layer aligns with conventional frame sampling methods [32]. As N increases, temporal information becomes more pronounced within each frame, though spatial detail decreases per frame. This trade-off enables the TIME layer to enhance temporal dynamics selectively while retaining essential spatial context.

3.2. TIME Layer Across Models

Our experiments apply the TIME layer to a range of foundational architectures for action recognition: CNN-based models (*e.g.*, ResNet-50), Transformer-based models (*e.g.*,

ViT), and self-supervised video models (*e.g.*, VideoMAE). These architectures represent key approaches in frame-based and video-based learning, allowing us to explore the TIME layer's adaptability in capturing temporal dynamics.

ResNet-50 with the TIME Layer. Incorporating the TIME layer into ResNet-50 enables it to handle temporal dependencies directly within frame-based processing, enhancing spatial patterns by embedding long-term and short-term temporal cues into each frame. This allows the CNN, typically optimized for spatial tasks (*e.g.*, image classification), to improve in tasks that require motion sensitivity.

ViT with the TIME Layer. For ViT, a model structured for efficient spatial patch processing, the TIME layer introduces spatio-temporal patches, capturing both short- and long-term temporal dynamics without altering ViT's framework. By encoding long-term motion through uniform sampling and capturing immediate motion continuity per frame, the TIME layer enhances ViT's utility for video data.

VideoMAE with the TIME Layer. In self-supervised contexts, the TIME layer integrates naturally with VideoMAE's tube masking, enabling it to handle both short-term and long-term dynamics. Extending VideoMAE's capabilities to depth video, we demonstrate how the TIME layer's integration across different data modalities provides comprehensive spatio-temporal information, making it valuable for tasks involving complex action dynamics.

These applications illustrate the TIME layer's flexibility across varied model architectures and data types, confirming its potential as a universal module in video action recognition. By enriching models with temporal awareness and supporting diverse modalities, the TIME layer offers a solid pathway toward more accurate and adaptable action recognition systems. Below, we provide key insights into how the TIME layer enhances video action recognition.

3.3. The Role of TIME Layer

Adapting image models for video tasks. The TIME layer's integration with standard image classifiers like ResNet-50 and ViT transforms them into effective video processors without requiring major architectural changes. By enriching frames with temporal information, the TIME layer equips these models to handle complex motion cues essential for action recognition, bridging the gap between image-based architectures and video data requirements.

Adjustable spatial-temporal balance. Using an $N \times N$ grid structure, the TIME layer balances spatial and temporal details within each frame, controlled by N. Smaller N values emphasize spatial clarity, preserving fine visual details, while larger values incorporate greater temporal dynamics, revealing motion over time. This adjustable framework enables researchers to finely tune the spatial-temporal balance, offering insights into model sensitivity to these factors and supporting deeper analysis of action recognition needs.

Datasets	Classes	Subjects	Views	Video clips	Modalities
MSRAction3D [17]	20	10	1	567	Depth
3D Action Pairs [20]	12	10	1	360	RGB+Depth
UWA3D Activity [22]	30	10	1	701	RGB+Depth
UWA3D Multiview Activity II [23]	30	9	4	1,070	RGB+Depth
HMDB51 [16]	51	-	-	6,766	RGB
UCF101 [27]	101	-	-	13,320	RGB
NTU RGB+D [25]	60	40	80	56,880	RGB+Depth
NTU RGB+D 120 [19]	120	106	155	114,480	RGB+Depth

Table 1. We evaluate both RGB and depth videos across a variety of datasets, chosen to represent a wide range of action recognition challenges. These datasets vary from small-scale to large-scale, complex benchmarks, which involve long sequences and intricate temporal dynamics. They cover difficulties such as motion ambiguity, cluttered backgrounds, viewpoint variations, low resolution, and diverse action types, all requiring models that can handle complex temporal and spatial dependencies.

Diagnostic tool for model interpretation. The TIME layer functions as a diagnostic tool, showcasing how models absorb and interpret spatial and temporal cues across various setups. By testing continuous sequences, temporally ordered grids, and combinations of short- and long-term dynamics, the TIME layer provides insights into how CNNs, transformers, and self-supervised models like VideoMAE respond to detailed spatial-temporal inputs, enhancing our understanding of model interpretability and adaptability.

Systematic evaluation of spatial-temporal information. Unlike traditional methods that use sparse sampling or treat frames independently, the TIME layer uses the full video sequence, enriching temporal information within each spatial frame structure. This systematic approach allows for precise manipulation of spatial-temporal elements, providing a powerful tool to assess the critical balance of motion and visual cues in action recognition.

4. Experiment

This section presents our experiments, and evaluations, followed by an in-depth discussion on the TIME layer. In the Appendix, we include additional experimental results, visualizations, and in-depth discussions for further insights.

4.1. Setup

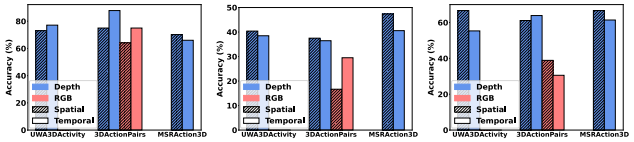
Models & datasets. In this work, we evaluate three representative models (Sec. 3.2), covering both frame-based and video-based architectures, across two modalities: conventional RGB and depth. Our analysis spans eight diverse datasets, ranging from small-scale to large-scale, which collectively present various challenging scenarios. These include datasets capturing sport-related activities, such as *jogging*, *side boxing*, and *golf swing* (*e.g.*, MSRAction3D [17]), where distinct motion characteristics are critical. Additionally, we evaluate on pairs of actions with highly similar motion trajectories (*e.g.*, *pick up a box* and *put down a box*; *stick a poster* and *remove a poster* in 3D

Action Pairs [20]), testing the models' ability to discern fine-grained differences. We further analyze performance on actions performed at various heights and speeds in cluttered scenes that feature frequent self-occlusions (UWA3D Activity [22]), and actions captured from multiple camera viewpoints (UWA3D Multiview Activity II [23]), examining the robustness to occlusion and viewpoint changes. The HMDB51 dataset [16] introduces challenges of low-resolution videos captured in uncontrolled settings, emphasizing motion ambiguity, viewpoint variations, and differing pose orientations that are common in real-world applications. The UCF101 dataset [27] further highlights issues in actions with inherent motion ambiguity, variations in viewpoint, and diverse pose configurations. Lastly, we include two large-scale RGB+D datasets, NTU RGB+D (NTU-60) [25] and NTU RGB+D 120 (NTU-120) [19], containing longer sequences with complex temporal dynamics, which allow for an in-depth evaluation of the models' ability to understand and recognize actions with extended temporal structures. A summary of these datasets is in Tab. 1.

Implementations & metrics. Since our TIME layer generates new frame images by rearranging the original video frames, these frames differ from typical real images (*e.g.*, ImageNet-1K [7]) in terms of both visual content and layout. The generated frames contain repeated visual patterns with some motion, and while the scale of objects and subjects may differ, the visual content resembles that of ImageNet-1K. As a result, no existing datasets with image-level labels are directly applicable for pretraining on these transformed frames. Given this, we focus on fine-tuning ImageNet-1K pretrained models for our experiments. For fine-tuning, we use ResNet-50 [13] and ViT [8] models, both pretrained on ImageNet-1K, using two V100 GPUs.

For pretraining VideoMAE on the datasets listed in Tab. 1, we use the base model with a patch size of 16 and an input resolution of 224×224. Training is conducted on four V100 GPUs for all datasets. Following the original paper, we use the AdamW optimizer with a 40-epoch warm-up phase. We adjust the batch size based on the dataset, ranging from 24 to 128 (*e.g.*, 128 for both HMDB51 and UCF101), with the total number of epochs set between 80 and 4800. For instance, NTU-60 and NTU-120 are trained for 200 epochs [39], UCF101 for 3200 epochs, and HMDB51 for 4800 epochs [28], with the number of epochs determined by dataset size and complexity. During fine-tuning, we follow the procedure outlined in [28], using a base learning rate of 5e-4 for UCF101 and 1e-3 for HMDB51, with a weight decay of 0.7 and a batch size of 128 for both datasets. Fine-tuning is performed for 50 epochs on HMDB51 and 100 epochs on UCF101. For the large-scale NTU-60 and NTU-120 datasets, we perform fine-tuning for 50 epochs.

We primarily report Top-1 recognition accuracy, and in some cases, to deepen the evaluation, we also include Top-5 performance. Additionally, we analyze the effects of the TIME layer by measuring cosine similarity between model weights, layer by layer, comparing models trained with and without the TIME layer. This comparison provides insights into the influence of the TIME layer across different datasets, video modalities, mask ratios in VideoMAE, and both from-scratch training and fine-tuning setups.



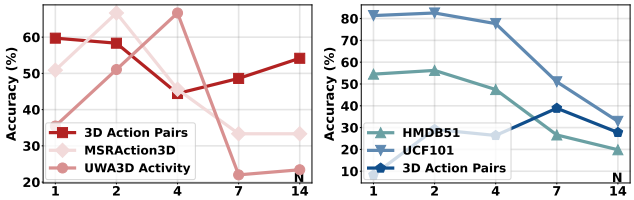
(a) ResNet-50. (b) ViT. (c) VideoMAE.

Figure 2. Comparison of spatial and temporal block arrangements across RGB and depth modalities using ResNet-50 (pretrained on ImageNet-1K), ViT (pretrained on ImageNet-1K), and VideoMAE (trained from scratch) on three datasets.

4.2. Analysis and Evaluation

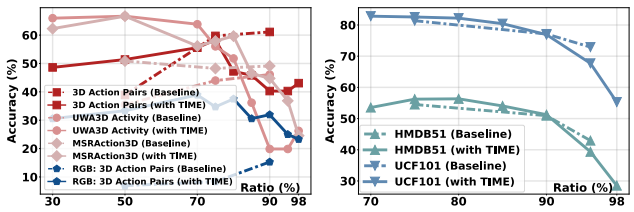
Spatial vs. temporal block arrangements. We conduct an ablation study on two TIME layer variants (described in Sec. 3.1), using ResNet-50 and ViT models fine-tuned with ImageNet-1K pretrained weights, and training VideoMAE from scratch. Results show that ViT benefits more from spatial block arrangement on depth videos, while temporal block arrangement excels on 3D Action Pairs (RGB), likely due to the dataset's action pairs featuring similar motion, where temporal block arrangement better captures short-term and subtle motions. For ResNet-50, the temporal block arrangement provides better performance, except on the MSRAction3D (Depth). This can be attributed to the pretrained ResNet-50's ability to capture fine visual details, which benefits from short-term motion captured per frame in the temporal block arrangement. With VideoMAE, spatial block arrangement performs on par or better across both modalities and datasets. Therefore, for the remaining evaluations, we use the spatial block arrangement.

N as spatial-temporal balance parameter. As shown in Fig. 3a, UWA3D Activity (Depth) benefits from a higher N (*e.g.*, $N=4$), likely because depth videos separate the foreground subject more clearly, providing sharper silhouettes even amidst cluttered scenes. In contrast, RGB datasets like HMDB51 and UCF101 achieve better performance with a smaller N (typically $N=2$, as in Fig. 3b), where emphasizing temporal information aids in recognizing actions with complex visual environments. Increasing N reduces performance, possibly due to the visual complexity and lower resolution. Notably, 3D Action Pairs (RGB), with its single, clearly defined subject and absence of significant motion ambiguity, performs better with a much higher N (*e.g.*, $N=7$), reflecting the simpler nature of its actions compared



(a) Depth modality. (b) RGB modality.

Figure 3. Impact of the spatial-temporal balance parameter N on VideoMAE. For the RGB modality, smaller values of N generally yield better performance, as RGB content tends to be more visually complex. In contrast, the 3D Action Pairs dataset, containing simpler actions, achieves optimal results with a larger N compared to more challenging datasets like HMDB51 and UCF101.



(a) On small-scale datasets. (b) On HMDB51 and UCF101.

Figure 4. Impact of the TIME layer on VideoMAE mask ratios. We analyze RGB and depth videos across various datasets, including small-scale and challenging ones like HMDB51. Results show that the TIME layer generally favors a lower mask ratio for optimal performance on RGB videos.

to HMDB51 and UCF101, where pose, viewpoint, and motion variability are high. These findings suggest that the optimal N setting aligns with the dataset's visual structure and complexity, with simpler or well-segmented scenes benefitting from a higher spatial-temporal balance parameter.

Impact on VideoMAE mask ratios. As shown in Fig. 4a, smaller datasets in the RGB modality generally perform best with higher mask ratios (*e.g.*, 70–80% for 3D Action Pairs), while depth modalities tend to favor lower mask ratios, *e.g.*, 50% for MSRAction3D and UWA3D Activity. For more challenging datasets (as shown in Fig. 4b), an optimal mask ratio of approximately 80% is observed for HMDB51 and UCF101. Increasing the mask ratio beyond this point leads to a dramatic decrease in performance. This shows that the TIME layer's integration of temporal information benefits from a lower mask ratio, which retains sufficient visual details to enhance feature extraction effectively.

TIME layer in frame- and video-level models. As shown in Tab. 2, with the TIME layer, VideoMAE generally outperforms frame-level models such as ViT and ResNet-50. This is because VideoMAE is more effective at processing video data through mechanisms like tube masking and reconstructing missing spatiotemporal cues. With the addition of the TIME layer, VideoMAE further enhances its ability to extract spatial-temporal information, resulting in consistent performance improvements, particularly for training-from-scratch setups. When fine-tuned with the TIME layer on small datasets, the ImageNet-1K pre-trained models (ViT and ResNet-50) also demonstrate strong performance. This demonstrates that the TIME layer enhances frame-level models for video tasks by effectively injecting temporal information, striking a balance between spatial and temporal features. Interestingly, we also observe that ResNet-50, as a frame-level model with the TIME layer, achieves competitive results compared to VideoMAE. This may be due to a spatial bias in existing methods, as noted in [12, 21], where a single frame can sometimes be sufficient to recognize actions in Kinetics-400 without requiring the full sequence. These findings underscore the need for datasets that demand more robust temporal reasoning to fully evaluate video-level understanding.

TIME layer's effect across modalities. As shown in Tab. 2, the depth modality consistently outperforms conventional RGB across nearly all backbones. Depth videos enable easier foreground segmentation of human subjects, even in cluttered scenes, since the absence of color eliminates distractions like clothing color. This allows action recognition models to focus on extracting high-level features that describe the action, rather than handling low-level segmentation tasks. Additionally, both ImageNet-1K and Kinetics-400 pretrained models perform effectively when fine-tuned on depth videos, often surpassing their performance on RGB video fine-tuning. This suggests that the inherent clarity and reduced noise in depth data can positively impact model performance, underscoring the role of data quality in action recognition.

TIME layer on the effect of dataset volume. As shown in Tab. 2, we notice that large-scale pretraining, *e.g.*, on Kinetics-400 and then finetuning on smaller datasets generally achieve better performance compared to the training from scratch using smaller datasets. This is mainly because smaller datasets have limited motion concepts, visual appearances and scenarios, whereas large-scale datasets provide more rich information. Even for the depth videos, we observe the same trend. We also notice that ResNet-50 pretrained on ImageNet-1K, achieve quite competitive results on all the datasets and modalities.

TIME layer on multiview action recognition. Tab. 3 shows results on UWA3D Multiview Activity II using VideoMAE pretrained on Kinetics-400 and fine-tuned, with and without the TIME layer, on both RGB and depth modalities. Due to the smaller size of this dataset, we focus solely on fine-tuning rather than training from scratch. We find that incorporating the TIME layer consistently improves accuracy across all 12 distinct pairs of training and testing viewpoints, yielding gains of around 10% for depth videos and 5% for RGB videos, highlighting the TIME layer's ef-

Backbone	Pretraining	Fine-tuning/Testing	Modality	TIME layer	Top-1	Top-5
ViT	ImageNet-1K	MSRAction3D	Depth	X	13.16	57.90
				✓ ($N = 7$)	**40.57**	**82.24**
		3D Action Pairs	Depth	X	22.22	85.42
				✓ ($N = 7$)	**36.46**	**97.22**
			RGB	X	14.24	57.29
				✓ ($N = 4$)	**29.51**	**94.10**
		UWA3D Activity	Depth	X	10.99	58.16
				✓ ($N = 7$)	**38.48**	**74.29**
		HMDB51	RGB	X	47.09	76.44
				✓ ($N = 2$)	**47.29**	**77.17**
		UCF101	RGB	X	**84.26**	**95.80**
				✓ ($N = 2$)	79.81	95.27
ResNet-50	ImageNet-1K	MSRAction3D	Depth	X	17.98	61.40
				✓ ($N = 7$)	**66.01**	**94.96**
		3D Action Pairs	Depth	X	42.36	98.97
				✓ ($N = 14$)	**87.85**	**100.00**
			RGB	X	47.57	**100.00**
				✓ ($N = 4$)	**75.00**	**100.00**
		UWA3D Activity	Depth	X	64.01	93.62
				✓ ($N = 4$)	**77.13**	**97.52**
		HMDB51	RGB	X	**45.05**	**76.54**
				✓ ($N = 2$)	42.83	73.17
		UCF101	RGB	X	**76.55**	**93.79**
				✓ ($N = 2$)	71.27	92.02
VideoMAE	Kinetics-400	MSRAction3D	Depth	X	70.18	**98.25**
				✓ ($N = 2$)	**73.68**	97.37
	MSRAction3D	MSRAction3D	Depth	X	50.88	87.72
				✓ ($N = 2$)	**66.67**	**93.86**
	Kinetics-400	3D Action Pairs	Depth	X	79.17	**100.00**
				✓ ($N = 2$)	**82.34**	**100.00**
			RGB	X	**72.22**	**100.00**
				✓ ($N = 2$)	66.88	97.98
	3D Action Pairs	3D Action Pairs	Depth	X	**61.11**	**100.00**
				✓ ($N = 2$)	58.33	**100.00**
			RGB	X	**38.89**	**100.00**
				✓ ($N = 7$)	**38.89**	94.44
	Kinetics-400	UWA3D Activity	RGB	X	**88.80**	**100.00**
				✓ ($N = 2$)	84.38	**100.00**
			Depth	X	**81.56**	**99.29**
				✓ ($N = 2$)	81.25	97.92
	UWA3D Activity	UWA3D Activity	Depth	X	46.10	91.49
				✓ ($N = 4$)	**66.67**	**96.45**
	HMDB51	HMDB51	RGB	X	54.51	82.97
				✓ ($N = 2$)	**56.33**	**82.99**
	UCF101	UCF101	RGB	X	81.37	96.30
				✓ ($N = 2$)	**82.56**	**97.36**

Table 2. Evaluation of three representative backbones on conventional RGB and depth videos. For VideoMAE, we perform pretraining and finetuning on various datasets, while the other two backbones are finetuned from ImageNet-1K pretrained models. The evaluation is conducted with and without the TIME layer, indicated by (✓) and (X), respectively. Values in parentheses represent the optimal spatial-temporal balance parameters N, as defined in Sec. 3.1. For depth videos, larger values of N tend to improve performance, likely due to better foreground segmentation and the lack of color information, which emphasizes object and human silhouettes. Additionally, training from scratch with the TIME layer generally outperforms the baselines.

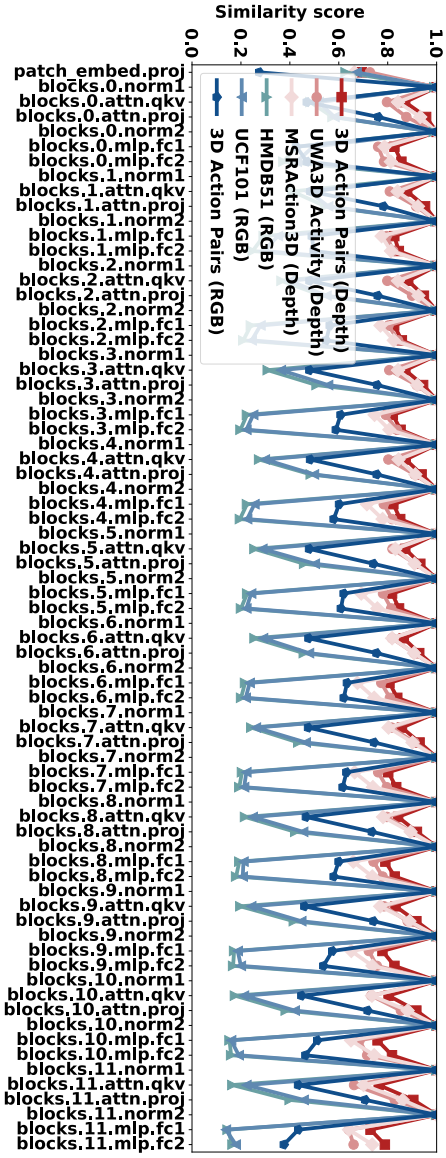


Figure 5. Per-layer weight similarity comparison between VideoMAE encoders with and without the TIME layer. The models are first trained from scratch, followed by finetuning of the encoders. The horizontal axis shows cosine similarity scores (ranging from 0 to 1), where higher values indicate greater similarity between the weights of the models. The vertical axis lists the individual layers within the VideoMAE encoder.

fectiveness in handling viewpoint variations and motion details. Notably, the TIME layer enables fine-tuning of the depth modality even when starting from RGB-pretrained models, demonstrating its versatility across modalities.

Using TIME layer as a diagnostic tool. We use cosine similarity to compare the per-layer weights of VideoMAE encoders between finetuned models that are firstly trained from scratch, with and without the TIME layer, across RGB and depth modalities. Notably, for all RGB modality datasets, we observe that the similarity scores for attention

	Training	V_1 & V_2		V_1 & V_3		V_1 & V_4		V_2 & V_3		V_2 & V_4		V_3 & V_4		Average
	Testing	V_3	V_4	V_2	V_4	V_2	V_3	V_1	V_4	V_1	V_3	V_1	V_2	
Depth	w/o K400 tuning	70.52	62.17	67.67	63.67	58.27	63.43	64.31	57.30	63.94	66.42	69.89	67.29	64.73
	w/ K400 tuning	79.48	76.40	75.19	76.40	70.30	74.25	76.95	70.41	78.44	76.49	76.21	73.31	75.06
	+ TIME layer	**89.18**	**85.02**	**87.55**	**83.52**	**78.57**	**79.48**	**86.99**	**77.15**	**89.22**	**88.81**	**89.96**	**86.47**	**85.16**
RGB	w/o K400 tuning	84.76	84.39	88.06	84.39	76.12	77.32	89.96	84.39	88.48	82.53	91.08	86.94	84.59
	+ TIME layer	**91.45**	**89.96**	**91.04**	**89.59**	**82.09**	**85.50**	**91.08**	**89.22**	**88.48**	**87.73**	**92.19**	**90.67**	**89.08**

Table 3. Evaluations on UWA3D Multiview Activity II show that the TIME layer consistently enhances performance in VideoMAE, both with and without Kinetics-400 (K400) fine-tuning, across RGB and depth modalities in all 12 training and testing viewpoint combinations.

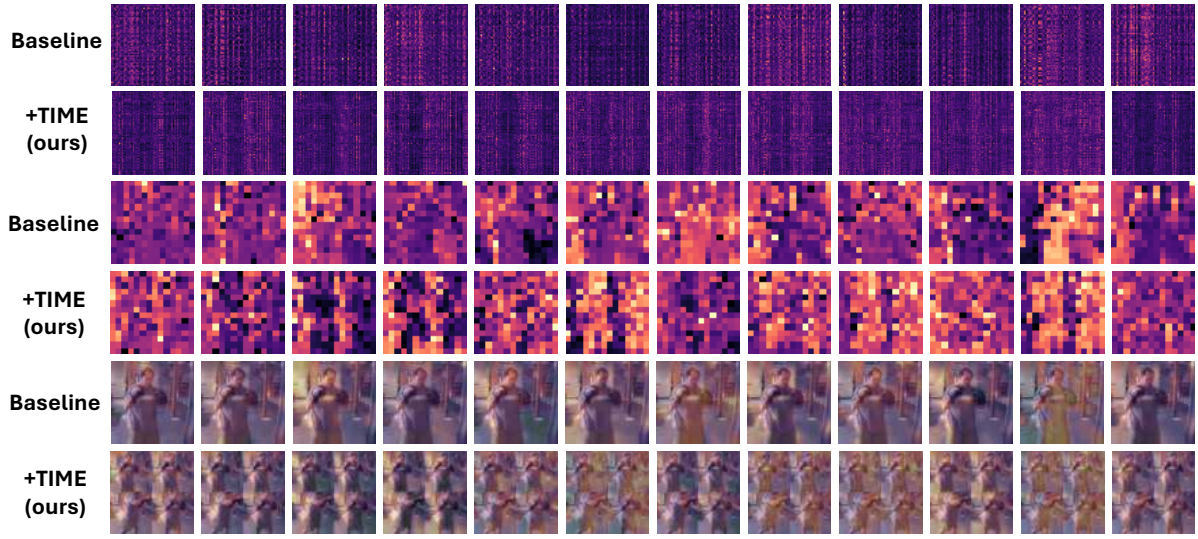


Figure 6. Visualization of attention maps, class weight heatmaps, and overlaid attention maps on VideoMAE with and without the TIME layer (Baseline vs. +TIME). The first two rows show attention maps from all 12 layers of the VideoMAE encoder, capturing the spatial distribution of attention across video frames and highlighting areas of focus at each layer. The third and fourth rows depict class weight heatmaps, derived from averaging attention scores of the [CLS] token across a 14×14 grid of spatial patches. These heatmaps indicate which patches are most relevant for classification, with the TIME layer (+TIME) producing sharper, more distinct attention regions compared to the Baseline. The last two rows overlay class weight heatmaps onto original video frames, showing that the TIME layer enables a more focused attention on critical regions, such as central subjects and areas of motion, while the Baseline displays a more diffused focus. This demonstrates how the TIME layer enhances the model's capacity to capture key features relevant for action classification.

projection layers are smaller. This suggests that the TIME layer influences the weights of fully connected layers responsible for embedding temporal information, enriching video processing tasks with nuanced motion cues.

In contrast, for depth datasets, the similarity scores tend to be higher compared to RGB datasets. Although the TIME layer still affects the attention projection layers, this result indicates that depth videos offer cleaner and more distinct silhouette information, which may reduce the impact of temporal adjustments. Our findings suggest that the TIME layer can serve as a valuable tool to assess model behavior and identify challenging datasets. For example, datasets like HMDB51 and UCF101 show much lower similarity scores for attention layers, indicating potential difficulties in processing complex actions with high viewpoint, pose, and motion variability. This insight can help fine-tune model performance and inform dataset selection.

Fig. 6 provides a visual comparison of attention maps generated with and without the use of the TIME layer. The TIME layer enhances attention maps by sharpening focus on key regions, especially areas with significant motion and central subjects, which are often critical for action recognition. This indicates that the TIME layer effectively directs the model's attention to relevant features, improving spatial precision and temporal sensitivity.

5. Conclusion

We introduced the TIME layer, a module designed to enhance temporal dynamics in video action recognition models with minimal architectural adjustments. By providing a flexible spatial-temporal balance, the TIME layer enables

both short-term and long-term temporal dynamics, resulting in improved model performance on complex action recognition tasks. Results show that the TIME layer effectively enriches temporal cues while preserving spatial fidelity, outperforming traditional methods. Its adaptability across diverse models, including ResNet-50, ViT, and VideoMAE, positions the TIME layer as a scalable and powerful solution for advancing future video processing research.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6836–6846, 2021. 2

[2] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *ICML*, page 4, 2021. 1, 2

[3] Hakan Bilen, Basura Fernando, Efstratios Gavves, Andrea Vedaldi, and Stephen Gould. Dynamic image networks for action recognition. In *CVPR*, 2016. 3

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1, 2

[5] Qixiang Chen, Lei Wang, Piotr Koniusz, and Tom Gedeon. Motion meets attention: Video motion prompts. In *The 16th Asian Conference on Machine Learning (Conference Track)*, 2024. 1, 2

[6] Xiyang Dai, Bharat Singh, Guyue Zhang, Larry S Davis, and Yan Qiu Chen. Temporal context network for activity localization in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5793–5802, 2017. 3

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 1, 5

[9] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 203–213, 2020. 3

[10] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016. 1

[11] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, pages 6202–6211, 2019. 2

[12] Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Omnimae: Single model masked pretraining on images and videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10406–10417, 2023. 1, 6

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 2, 5

[14] Kiyoon Kim, Shreyank N Gowda, Oisin Mac Aodha, and Laura Sevilla-Lara. Capturing temporal information in a single frame: Channel sampling strategies for action recognition. In *BMVC*. BMVA Press, 2022. 3

[15] Piotr Koniusz, Lei Wang, and Anoop Cherian. Tensor representations for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(2):648–665, 2021. 2

[16] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. HMDB: A large video database for human motion recognition. In *ICCV*, pages 2556–2563, 2011. 4, 5

[17] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3D points. *CVPR Workshop*, pages 9–14, 2010. 4

[18] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7083–7093, 2019. 2

[19] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 4, 5

[20] Omar Oreifej and Zicheng Liu. HON4D: Histogram of Oriented 4D Normals for Activity Recognition from Depth Sequences. In *CVPR*, pages 716–723, 2013. 4, 5

[21] Mandela Patrick, Dylan Campbell, Yuki Asano, Ishan Misra, Florian Metze, Christoph Feichtenhofer, Andrea Vedaldi, and Joao F Henriques. Keeping your eye on the ball: Trajectory attention in video transformers. *Advances in neural information processing systems*, 34:12493–12506, 2021. 1, 2, 6

[22] Hossein Rahmani, Arif Mahmood, Du Q. Huynh, and Ajmal Mian. HOPC: Histogram of Oriented Principal Components of 3D Pointclouds for Action Recognition. In *ECCV*, pages 742–757, 2014. 4, 5

[23] Hossein Rahmani, Arif Mahmood, Du Q. Huynh, and Ajmal Mian. Histogram of Oriented Principal Components for Cross-View Action Recognition. *TPAMI*, pages 2430–2443, 2016. 4, 5

[24] Michael S. Ryoo, AJ Piergiovanni, Mingxing Tan, and Anelia Angelova. Assemblenet: Searching for multi-stream neural connectivity in video architectures. In *International Conference on Learning Representations*, 2020. 1

[25] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. *CVPR*, pages 1010–1019, 2016. 4, 5

[26] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014. 1

[27] Khurram Soomro, Amir Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *ArXiv*, abs/1212.0402, 2012. 4, 5

[28] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022. 1, 2, 5

[29] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1, 2

[30] Lei Wang and Piotr Koniusz. Self-supervising action recognition by statistical moment and subspace descriptors. In *Proceedings of the 29th ACM international conference on multimedia*, pages 4324–4333, 2021. 2

[31] Lei Wang and Piotr Koniusz. Flow dynamics correction for action recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3795–3799. IEEE, 2024. 2

[32] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 1, 2, 3

[33] Lei Wang, Piotr Koniusz, and Du Q Huynh. Hallucinating idt descriptors and i3d optical flow features for action recognition with cnns. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8698–8708, 2019. 1, 2

[34] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(11): 2740–2755, 2019. 2

[35] Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14549–14560, 2023. 1, 2

[36] Lei Wang, Ke Sun, and Piotr Koniusz. High-order tensor pooling with attention for action recognition. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3885–3889. IEEE, 2024. 1

[37] Lei Wang, Xiuyuan Yuan, Tom Gedeon, and Liang Zheng. Taylor videos for action recognition. In *Forty-first International Conference on Machine Learning*, 2024. 1, 2, 3

[38] Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Jilan Xu, Zun Wang, et al. Internvideo2: Scaling video foundation models for multimodal video understanding. *ECCV*, 2024. 1

[39] Sangmin Woo, Sumin Lee, Yeonju Park, Muhammad Adi Nugroho, and Changick Kim. Towards good practices for missing modality robust action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2776–2784, 2023. 5

[40] Ceyuan Yang, Yinghao Xu, Jianping Shi, Bo Dai, and Bolei Zhou. Temporal pyramid network for action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 591–600, 2020. 2

[41] Liyun Zhu, Lei Wang, Arjun Raj, Tom Gedeon, and Chen Chen. Advancing video anomaly detection: A concise review and a new dataset. In *The Thirty-eighth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. 1

When Spatial meets Temporal in Action Recognition

Supplementary Material

Backbone	Pretraining	Fine-tuning/Testing	Modality	TIME layer	Top-1
VideoMAE	UCF101	UCF101	RGB	✗	88.3
				✓ ($N = 2$)	**89.4**
	NTU-60	NTU-60	Depth	✗	91.7
				✓ ($N = 2$)	**95.0**
			RGB	✗	85.3
				✓ ($N = 2$)	**87.2**
	NTU-120	NTU-120	Depth	✗	88.0
				✓ ($N = 2$)	**92.5**
			RGB	✗	84.5
				✓ ($N = 2$)	**85.3**

Table 4. Evaluation on the UCF101 and large-scale NTU-60 and NTU-120 datasets. VideoMAE is trained from scratch and fine-tuned for the action recognition task. The evaluation compares performance with and without the TIME layer, denoted by (✓) and (✗), respectively. On UCF101, we train VideoMAE from scratch for 3200 epochs, followed by fine-tuning for an additional 100 epochs.

Mask ratio	**Top-1**	**Top-5**
70%	89.40	98.39
75%	88.77	98.28
80%	87.58	98.20
85%	85.99	97.73
90%	84.17	97.15
95%	75.63	93.50
98%	65.66	88.61

Table 5. Evaluation of the mask ratio on UCF101.

A. Evaluations on UCF101, NTU-60 and NTU-120

Below, we present the evaluations conducted on the UCF101 and large-scale NTU-60 and NTU-120 datasets.

As shown in Tab. 4, incorporating the TIME layer enhances performance across both the RGB and depth modalities on NTU-60 and NTU-120. With the TIME layer, we achieve a performance boost of approximately 4% on both NTU-60 and NTU-120 datasets compared to the baseline, for the depth modality.

On UCF101, we observe that pre-training VideoMAE for 3200 epochs consistently enhances performance, compared to the 800-epoch pre-training used in Tab. 2 of the main paper. Notably, the inclusion of the TIME layer further amplifies these improvements.

B. Additional Results on UCF101

Tab. 5 presents the evaluation of mask ratios on UCF101. Unlike the standard VideoMAE, incorporating the TIME layer achieves better performance with lower mask ratios, such as 70% and 75%.

C. Additional Results on UWA3D Multiview Activity II

As shown in Tab. 6, for depth videos, the inclusion of the TIME layer significantly enhances performance. Specifically, it achieves improvements of over 15% without Kinetics-400 pre-training and over 10% with pre-training. These results demonstrate the TIME layer's ability to substantially boost action recognition performance.

D. Per-layer Weight Similarity Comparison

Below, we present a detailed comparison of per-layer weight similarity between the baseline model (without the TIME layer) and the model with the TIME layer, using VideoMAE, ViT, and ResNet-50 pretrained on either Kinetics-400 or ImageNet-1K. This comparison is performed across both RGB and depth modalities on various datasets.

We observe that fine-tuning with our TIME layer, particularly on VideoMAE (Fig. 7), significantly influences the weights of the later encoder layers. This suggests that the TIME layer is effectively extracting higher-level, abstract features, likely capturing temporal information essential for action recognition tasks. The adaptation of these later layers to incorporate action-related features highlights the impact of our TIME layer in enhancing the model's ability to capture and use temporal dynamics.

Compared to Fig. 5 in the main paper, we observe that training from scratch has a more pronounced effect on the layer weights. This highlights the importance of incorporating the TIME layer during large-scale pretraining, as it has the potential to further enhance downstream motion-related tasks, such as action recognition and anomaly detection.

Interestingly, we observe that fine-tuning ViT with the TIME layer leads to significant changes in the weights of the earlier layers (Fig. 8), particularly the attention and projection layers. This suggests that ViT has a limited capacity to effectively capture and process temporal information.

Additionally, we observe that the TIME layer, when fine-tuning the ResNet-50 pretrained models, significantly influences multiple layers across both RGB and depth modalities (Fig. 9). This indicates that the TIME layer enables a 2D model to begin extracting spatial information, which is later used as temporal information for motion-related tasks. This provides a novel perspective on incorporating temporal information into traditional 2D CNNs, which have predominantly been used for image classification tasks. With the integration of the TIME layer, these models can now be

	Training	V_1 & V_2		V_1 & V_3		V_1 & V_4		V_2 & V_3		V_2 & V_4		V_3 & V_4		Average
	Testing	V_3	V_4	V_2	V_4	V_2	V_3	V_1	V_4	V_1	V_3	V_1	V_2	
Depth	w/o K400 tuning	70.52	62.17	67.67	63.67	58.27	63.43	64.31	57.30	63.94	66.42	69.89	67.29	64.73
	+ TIME layer	**84.04**	**76.18**	**83.15**	**79.26**	**76.69**	**77.74**	**80.16**	**72.70**	**84.60**	**82.50**	**84.76**	**80.59**	**80.20**
	w/ K400 tuning	79.48	76.40	75.19	76.40	70.30	74.25	76.95	70.41	78.44	76.49	76.21	73.31	75.06
	+ TIME layer	**89.18**	**85.02**	**87.55**	**83.52**	**78.57**	**79.48**	**86.99**	**77.15**	**89.22**	**88.81**	**89.96**	**86.47**	**85.16**

Table 6. Evaluations on UWA3D Multiview Activity II show that the TIME layer consistently enhances performance in VideoMAE, both with and without Kinetics-400 (K400) fine-tuning, in all 12 training and testing viewpoint combinations.

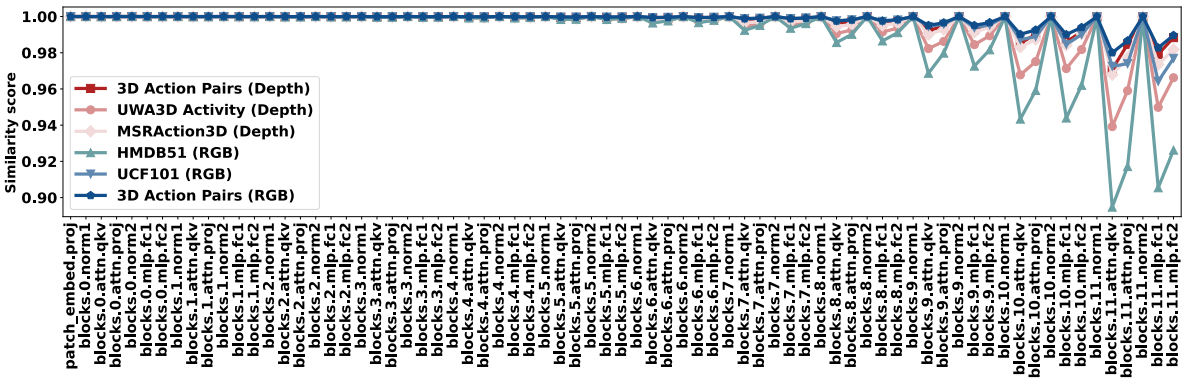


Figure 7. Comparison of per-layer weight similarity between the baseline model (without the TIME layer) and the model with the TIME layer, using VideoMAE pretrained on Kinetics-400 and fine-tuned across RGB and depth modalities on various datasets.

adapted to enhance video processing tasks, such as action recognition.

E. Additional Results on Spatial vs. Temporal Arrangements

We now present comparisons of per-layer weight similarity between spatial and temporal block arrangements on VideoMAE, ViT, and ResNet-50.

As illustrated in Fig.10, 11, and 12, the spatial and temporal block arrangements indeed lead to distinct weight patterns in all three different models.

In VideoMAE, both arrangements primarily influence the attention layers and some MLP layers. In contrast, on ViT, the effect is more pronounced in earlier layers, particularly attention and projection layers. Interestingly, on ResNet-50, the block arrangements impact nearly all convolutional layers. These observations suggest that VideoMAE excels at capturing both spatial and temporal information, while ResNet-50 is more focused on extracting spatial details. Overall, these results demonstrate that our TIME layer enables image-based models, such as ViT and ResNet-50, to be effectively adapted for video processing tasks.

We further visualize the attention / feature maps generated by both spatial and temporal block arrangements on VideoMAE, ViT, and ResNet-50, for both RGB and depth modalities.

As shown in Fig. 13 and Fig. 14, for the RGB modality, the spatial block arrangement tends to exhibit more concentrated attention. This could be attributed to the 3D Action Pairs dataset, which consists of action pairs with similar motion trajectories, resulting in more focused attention on specific regions where both motion and frame order are crucial. In contrast, for the depth modality, both spatial and temporal arrangements show similar attention patterns. This may be due to the depth videos being clearer and more compact compared to the RGB videos, which reduces the variability in attention across different arrangements.

In Fig. 17 and Fig. 18, we observe that for ResNet-50, both spatial and temporal block arrangements behave similarly. This is likely because ResNet-50, a model originally designed for image classification, excels at capturing spatial details, which reduces the difference in attention patterns between the two arrangements.

As shown in Fig. 15 and Fig. 16, when training VideoMAE from scratch, both spatial and temporal block arrangements exhibit similar attention patterns. This suggests that large-scale pretraining with the TIME layer in VideoMAE has the potential to improve performance on downstream tasks. VideoMAE excels at capturing both spatial and temporal information, and the TIME layer, acting as a bridge between these two modalities, further enhances the model's ability to extract spatio-temporal features essential for motion-related tasks.

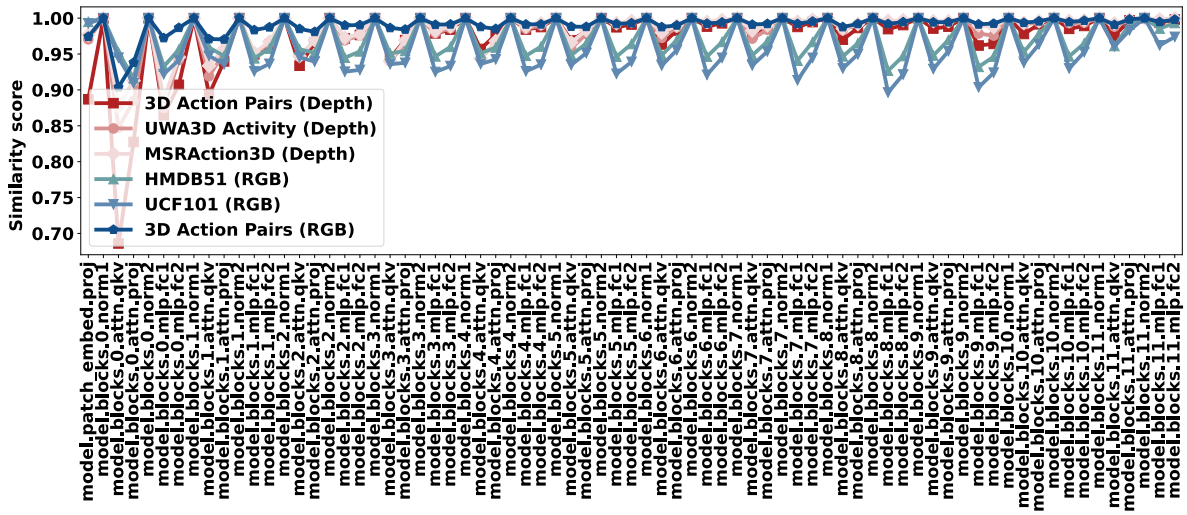


Figure 8. Comparison of per-layer weight similarity between the baseline model (without the TIME layer) and the model with the TIME layer, using ViT pretrained on ImageNet-1K and fine-tuned across RGB and depth modalities on various datasets.

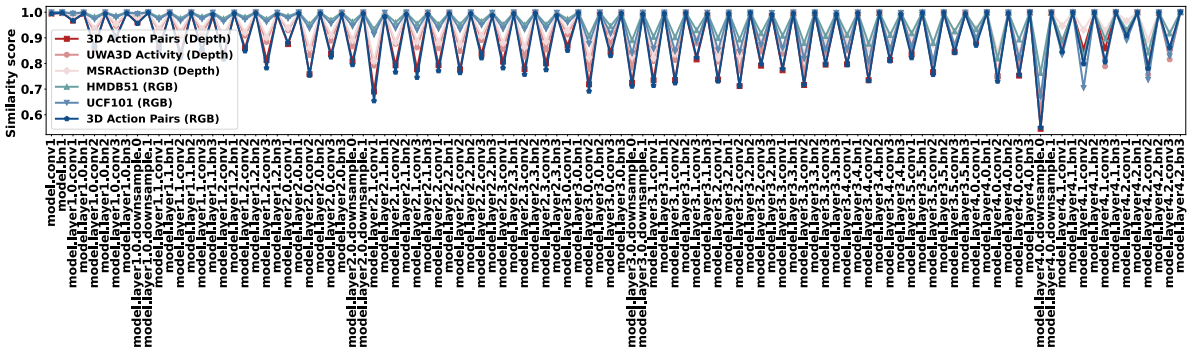


Figure 9. Comparison of per-layer weight similarity between the baseline model (without the TIME layer) and the model with the TIME layer, using ResNet-50 pretrained on ImageNet-1K and fine-tuned across RGB and depth modalities on various datasets.

F. Visualizations of Attention and Feature Maps

Fig. 19 to 27 present visual comparisons of attention and feature maps between the baseline (without the TIME layer) and the model with the TIME layer. These comparisons use different model architectures, including both training from scratch and fine-tuning, across various video modalities.

Overall, when applied to depth videos, the use of the TIME layer leads to more clear and compact attention maps and feature representations compared to those generated from RGB videos (*e.g.*, Fig. 20 vs. Fig. 21 for VideoMAE, Fig. 24 vs. Fig. 25 for ViT, Fig. 26 vs. Fig. 27). This enhancement is particularly noticeable in terms of how the model captures and organizes spatial and temporal information.

The attention maps from Vision Transformers (ViT) tend to exhibit weaker responses to temporal dynamics, struggling to fully capture the evolution of features across time. In contrast, the VideoMAE model demonstrates a more uniform focus across all regions of temporal dynamics, effectively maintaining attention throughout the video sequence (*e.g.*, Fig. 22 vs. Fig. 25 for depth, Fig. 20 vs. Fig. 24 for RGB).

When using a ResNet-50 backbone, depth video feature maps are generally more distinct and compact, showing clearer boundaries and less noise than those derived from RGB videos (*e.g.*, Fig. 26 vs. Fig. 27). In contrast, the feature maps from RGB videos often suffer from fuzzier, more ambiguous features with less defined boundaries.

In summary, the use of the TIME layer consistently improves the model's ability to capture temporal information across various architectures, from CNNs (*e.g.*, ResNet-50) to Transformers (*e.g.*, ViT), including self-supervised pretraining models like VideoMAE. This enhancement is observed for both RGB and depth modalities, underscoring

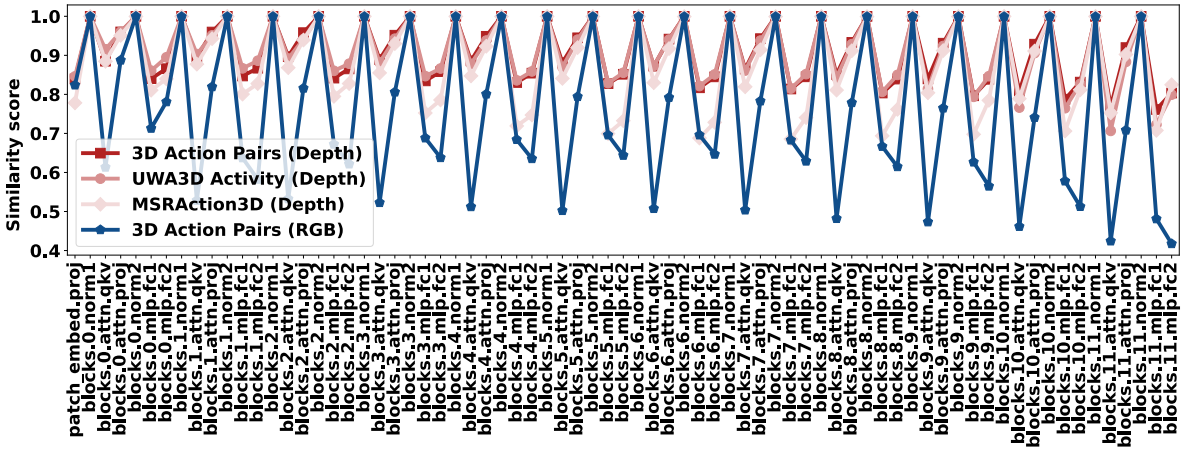


Figure 10. Comparison of per-layer weight similarity between spatial and temporal block arrangements using VideoMAE trained from scratch on three depth datasets (3D Action Pairs, UWA3D Activity, MSRAction3D) and one RGB dataset (3D Action Pairs).

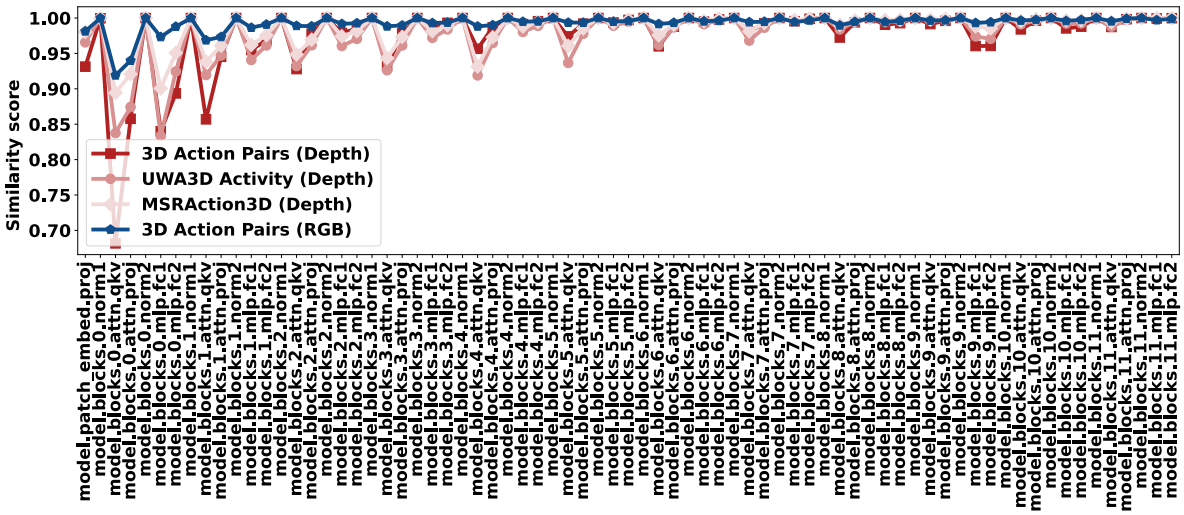


Figure 11. Comparison of per-layer weight similarity between spatial and temporal block arrangements using ViT pretrained on ImageNet-1K and fine-tuned on three depth datasets (3D Action Pairs, UWA3D Activity, MSRAction3D) and one RGB dataset (3D Action Pairs).

the TIME layer's ability to enhance temporal understanding regardless of the model architecture or input modality.

G. Visualizations of Sequence Reconstructions

Below, we present visualizations of our new video sequences with varying spatial-temporal balance parameters (N) and the corresponding reconstructed sequences generated by VideoMAE using masking ratios of 50%, 75%, and 90% for both RGB and depth modalities.

As illustrated in Fig. 28 to 37, the TIME layer effectively reconstructs temporal information in both RGB and depth video sequences. This highlights the TIME layer's ability to enhance temporal dynamics in video frames while simultaneously preserving spatial details.

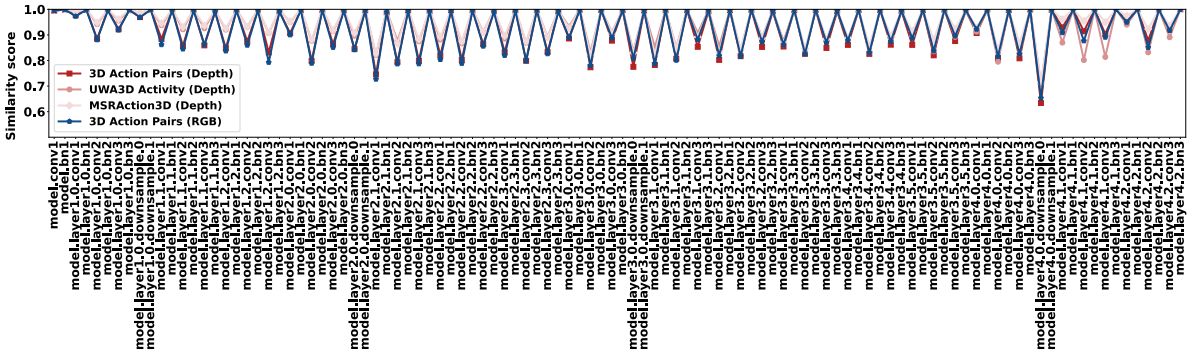


Figure 12. Comparison of per-layer weight similarity between spatial and temporal block arrangements using ResNet-50 pretrained on ImageNet-1K and fine-tuned on three depth datasets (3D Action Pairs, UWA3D Activity, MSRAction3D) and one RGB dataset (3D Action Pairs).

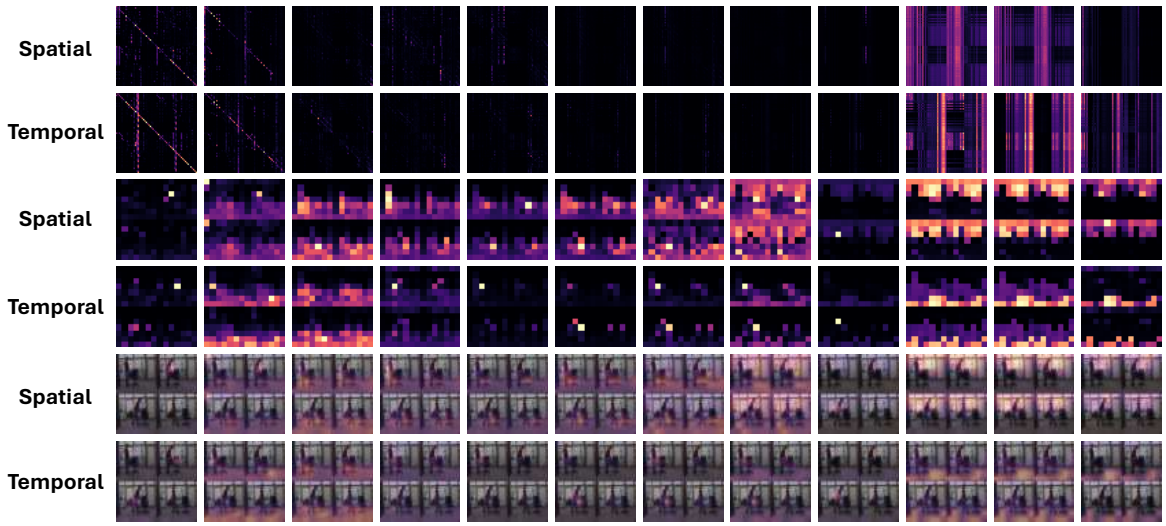


Figure 13. Visual comparison of attention maps for the action *push the chair*, between spatial and temporal block arrangements, using ViT (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs RGB).

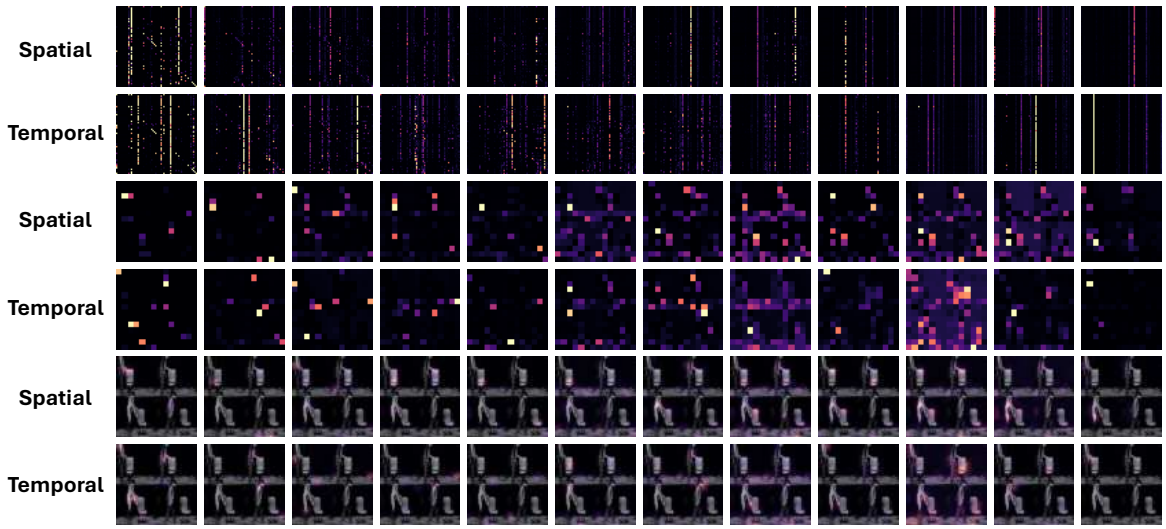


Figure 14. Visual comparison of attention maps for the action *push the chair*, between spatial and temporal block arrangements, using ViT (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs Depth).

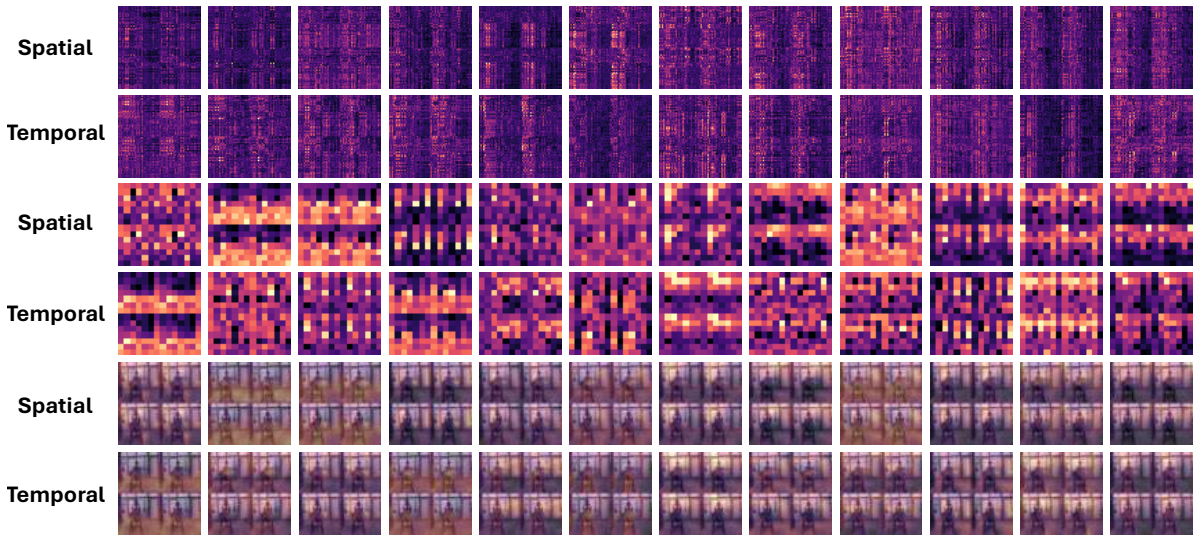


Figure 15. Visual comparison of attention maps for the action *put hat on*, between spatial and temporal block arrangements, using VideoMAE (trained from scratch on 3D Action Pairs RGB).

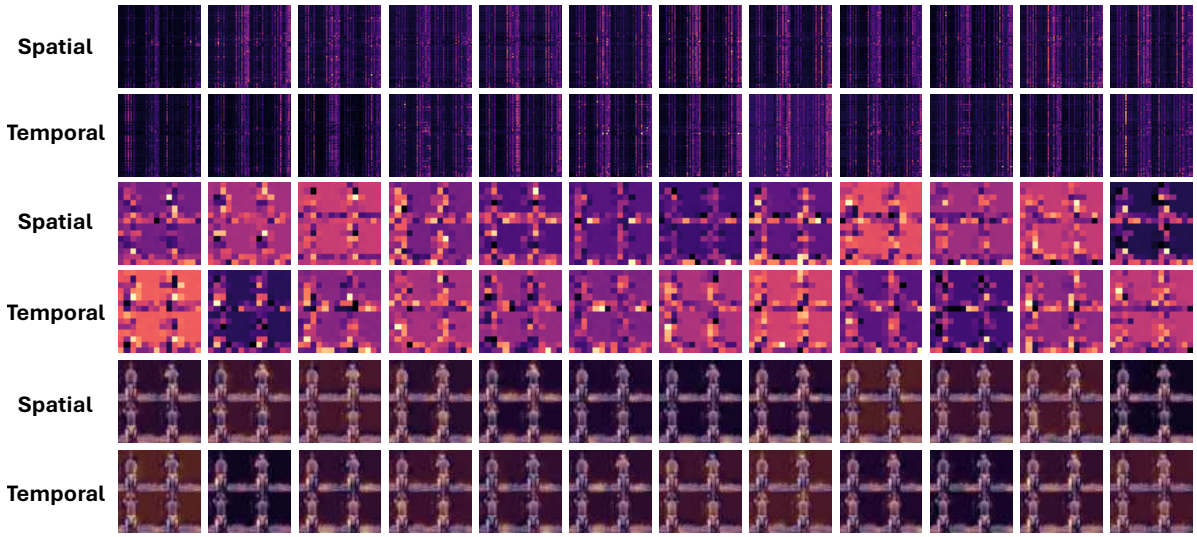


Figure 16. Visual comparison of attention maps for the action *put hat on*, between spatial and temporal block arrangements, using VideoMAE (trained from scratch on 3D Action Pairs Depth).

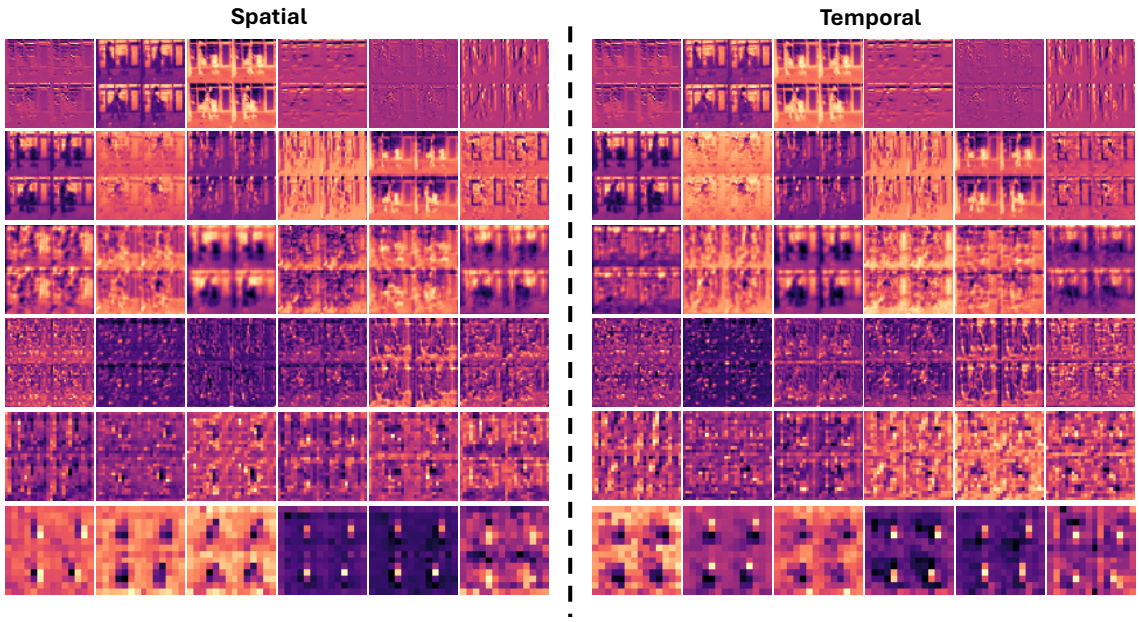


Figure 17. Visual comparison of attention maps for the action *push the chair*, between spatial and temporal block arrangements, using ResNet-50 (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs RGB).

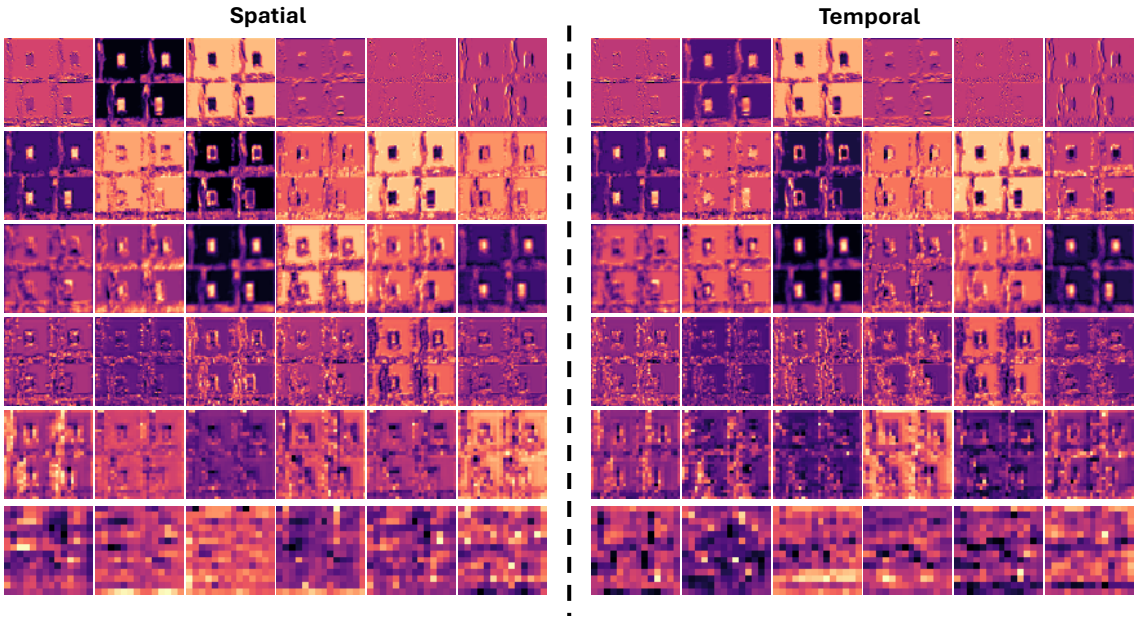


Figure 18. Visual comparison of attention maps for the action *push the chair*, between spatial and temporal block arrangements, using ResNet-50 (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs Depth).

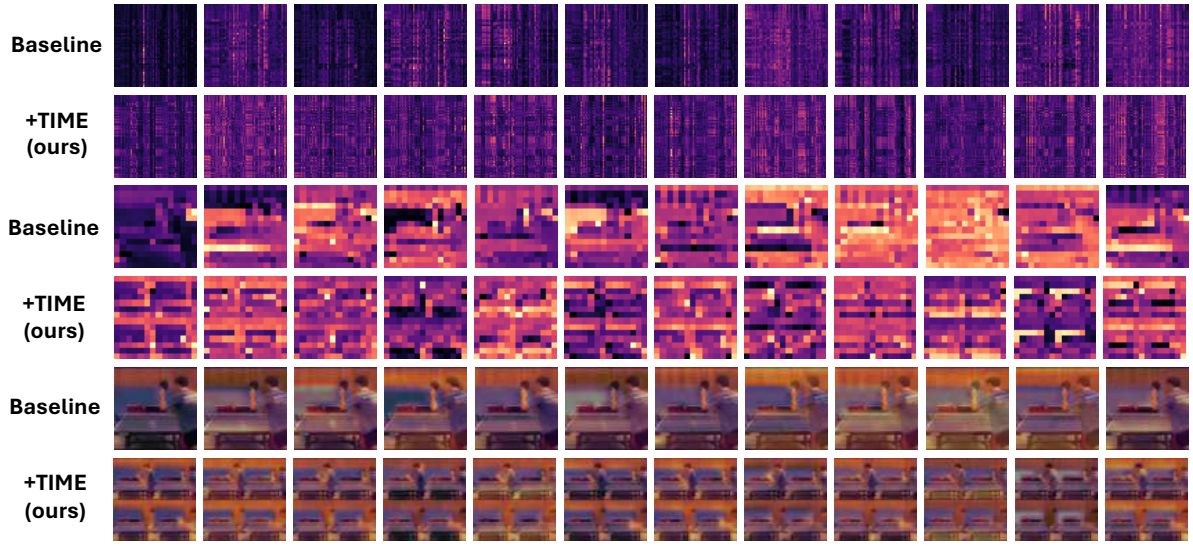


Figure 19. Visual comparison of attention maps for the action *table tennis*, between the baseline (without TIME layer) and the model with the TIME layer, using VideoMAE (train from scratch on UCF101 RGB).

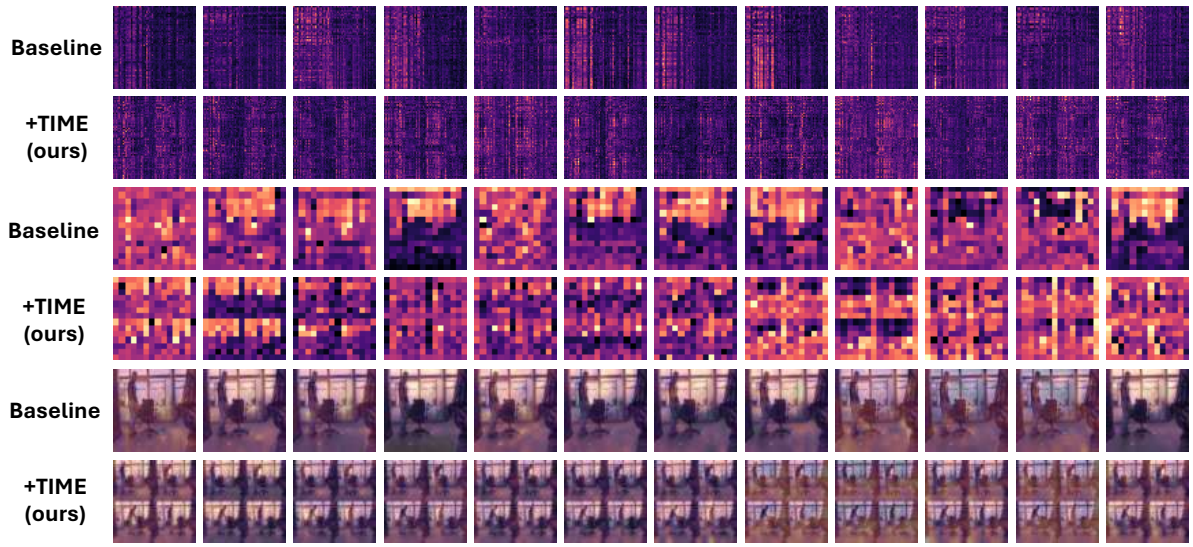


Figure 20. Visual comparison of attention maps for the action *push the chair*, between the baseline (without TIME layer) and the model with the TIME layer, using VideoMAE (train from scratch on 3D Action Pairs RGB).

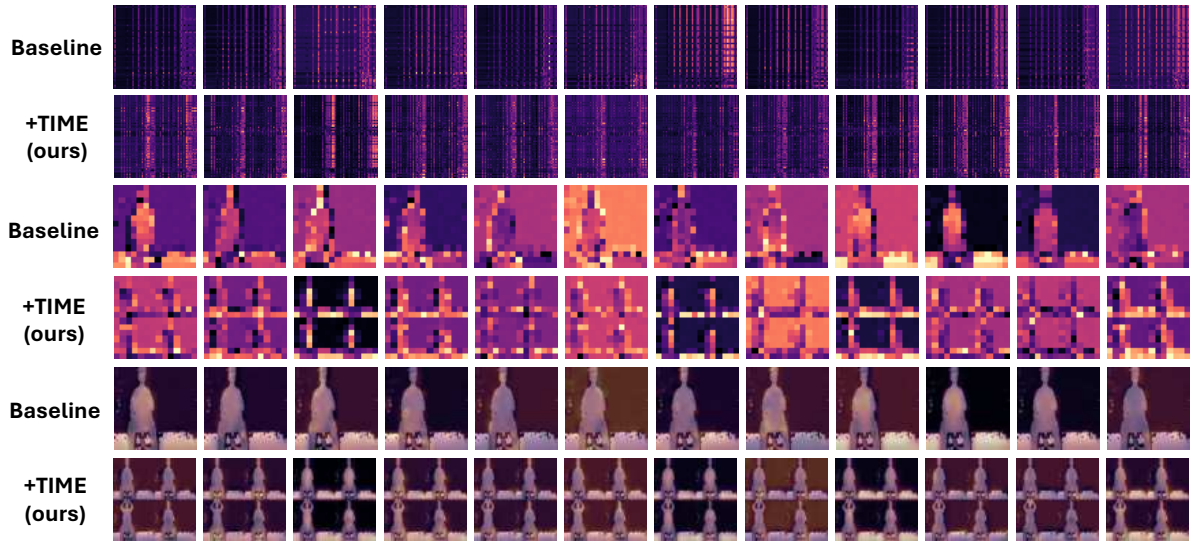


Figure 21. Visual comparison of attention maps for the action *take off hat*, between the baseline (without TIME layer) and the model with the TIME layer, using VideoMAE (train from scratch on 3D Action Pairs Depth).

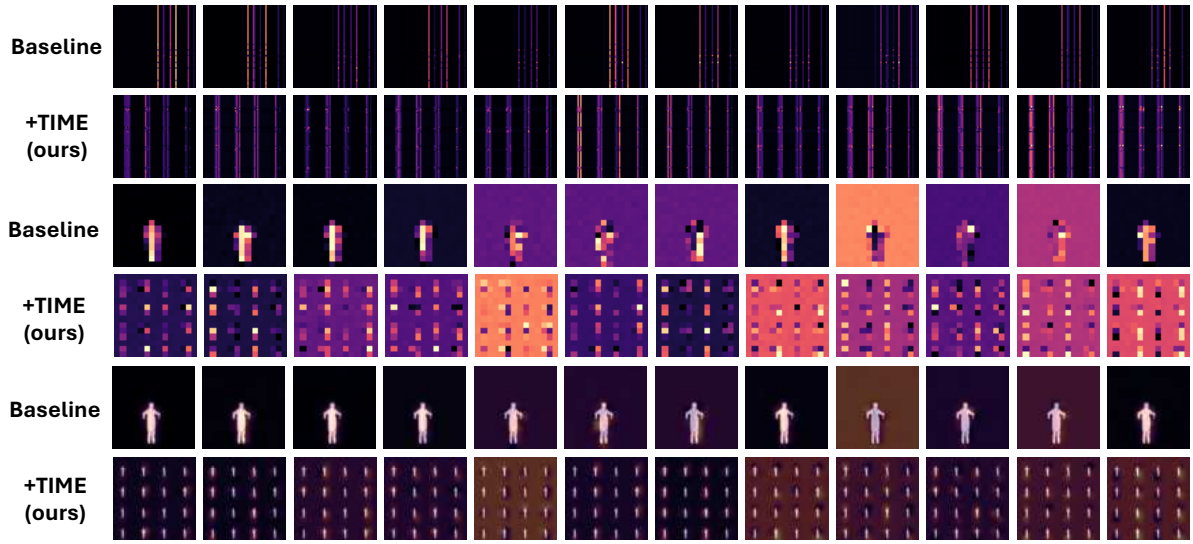


Figure 22. Visual comparison of attention maps for the action *hands on waist*, between the baseline (without TIME layer) and the model with the TIME layer, using VideoMAE (train from scratch on UWA3D Activity Depth).

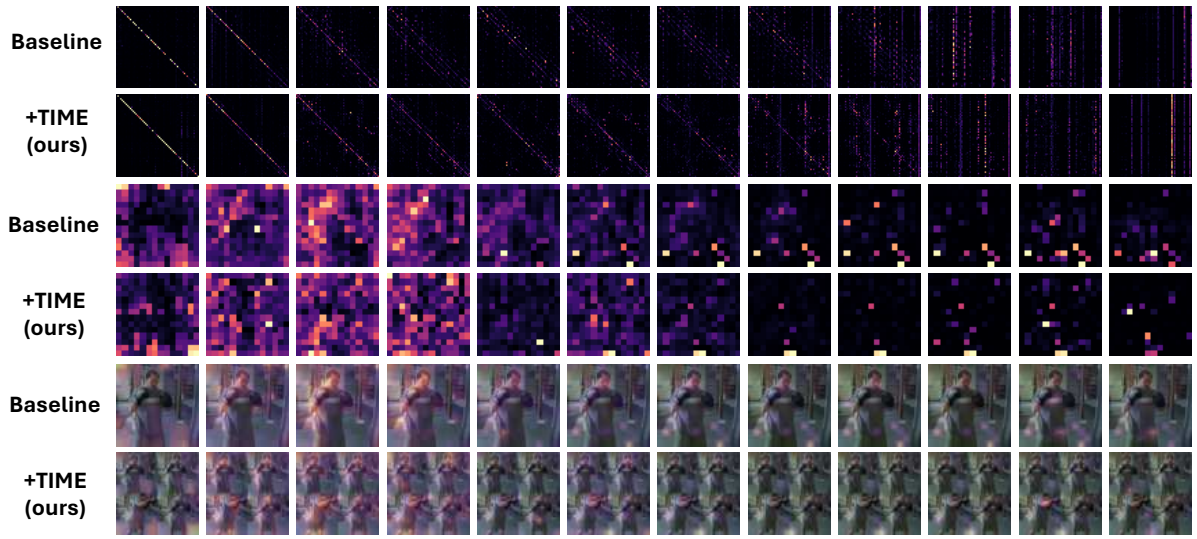


Figure 23. Visual comparison of attention maps for the action *boxing*, between the baseline (without TIME layer) and the model with the TIME layer, using ViT (pretrained on ImageNet-1K and fine-tuned on HMDB51 RGB).

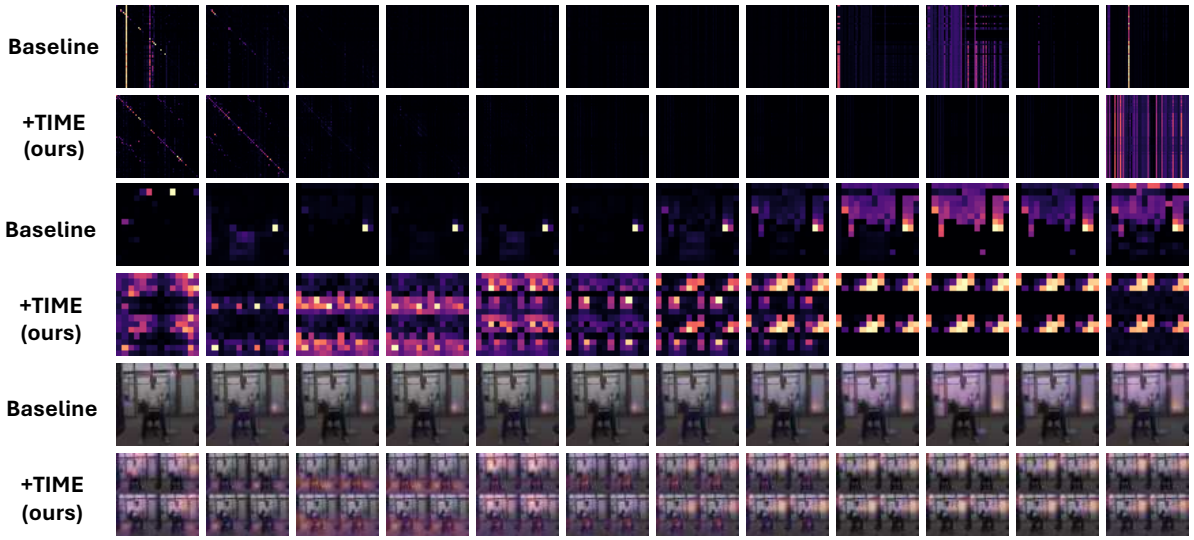


Figure 24. Visual comparison of attention maps for the action *put box on the desk*, between the baseline (without TIME layer) and the model with the TIME layer, using ViT (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs RGB).

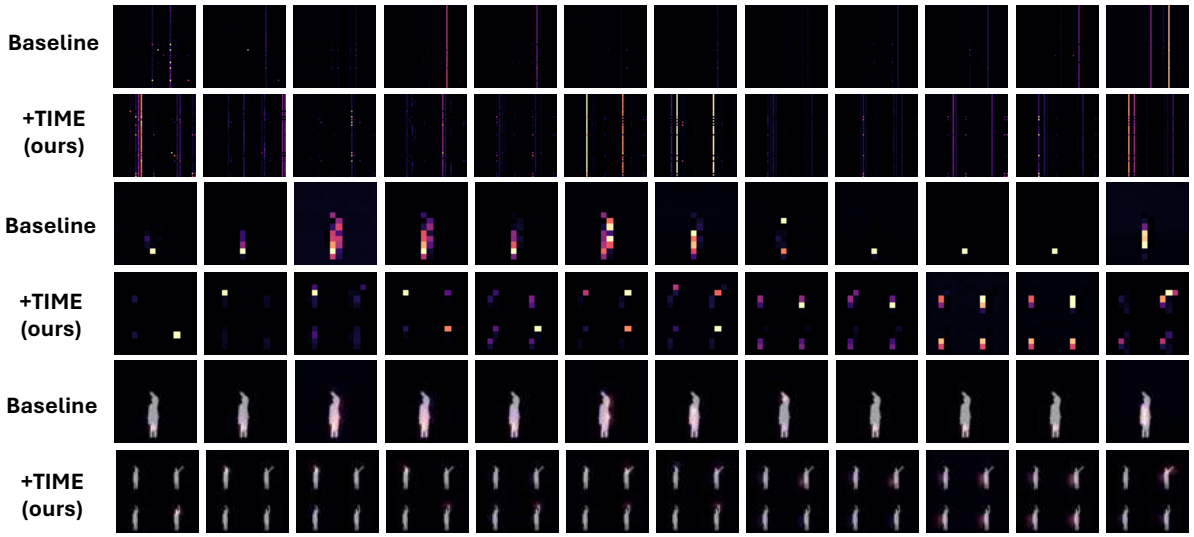


Figure 25. Visual comparison of attention maps for the action *high arm wave*, between the baseline (without TIME layer) and the model with the TIME layer, using ViT (pretrained on ImageNet-1K and fine-tuned on UWA3D Activity Depth).

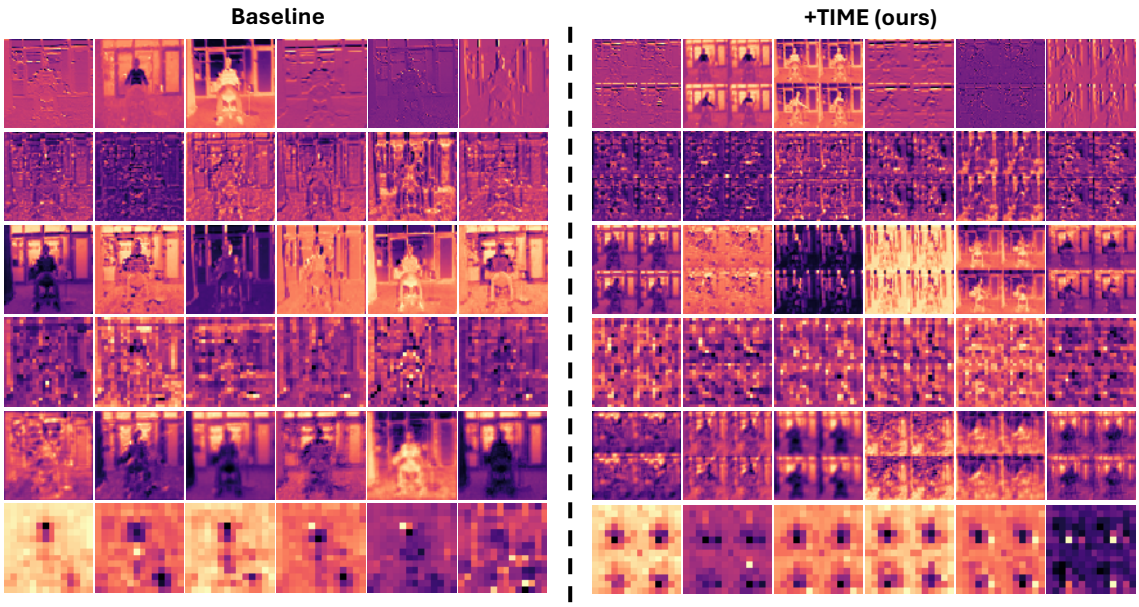


Figure 26. Visual comparison of feature maps for the action *pick up box on the desk*, between the baseline (without TIME layer) and the model with the TIME layer, using ResNet-50 (pretrained on ImageNet-1K and fine-tuned on 3D Action Pairs RGB). Each row displays the feature maps from a specific layer, listed from top to bottom in the following order: `conv1`, `layer1[0].conv1`, `layer1[0].conv2`, `layer2[0].conv1`, `layer3[0].conv1`, and `layer4[0].conv1`. One feature map per layer is shown.

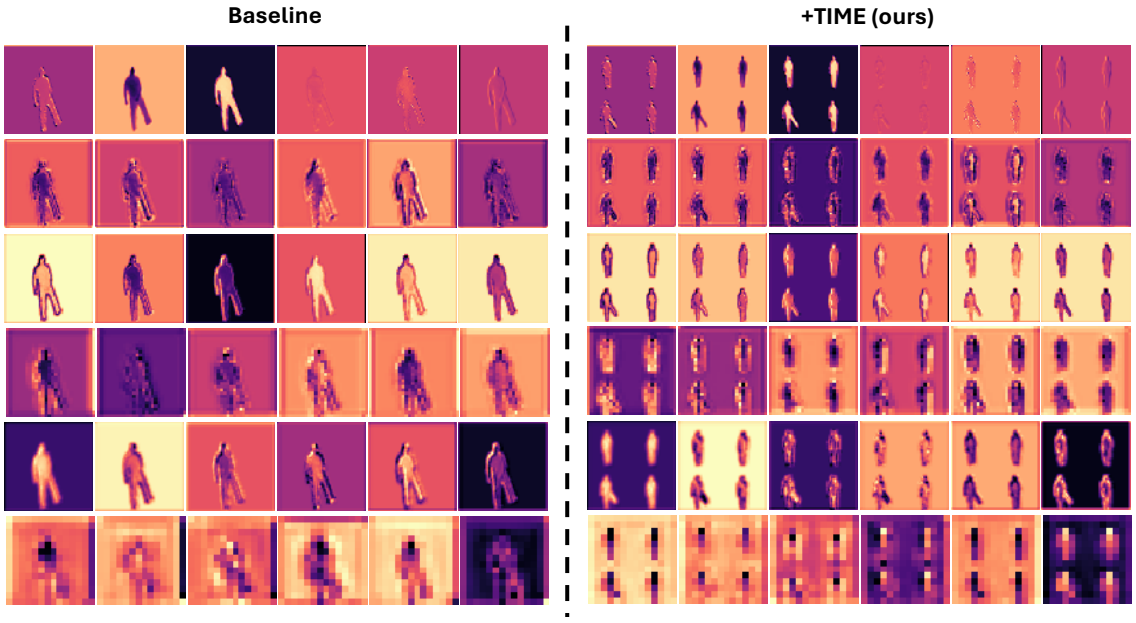


Figure 27. Visual comparison of feature maps for the action *side kick*, between the baseline (without TIME layer) and the model with the TIME layer, using ResNet-50 (pretrained on ImageNet-1K and fine-tuned on MSRAction3D Depth). Each row displays the feature maps from a specific layer, listed from top to bottom in the following order: `conv1`, `layer1[0].conv1`, `layer1[0].conv2`, `layer2[0].conv1`, `layer3[0].conv1`, and `layer4[0].conv1`. One feature map per layer is shown.

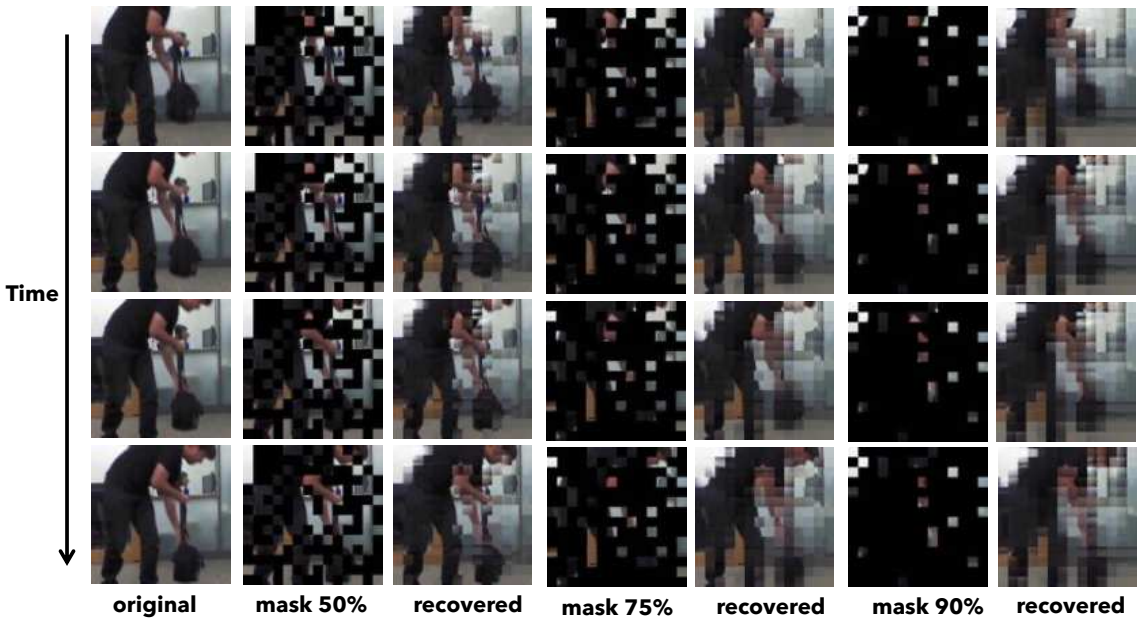


Figure 28. We present our new video sequence ($N = 1$, Baseline) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *put bag on the floor* from the 3D Action Pairs RGB dataset.

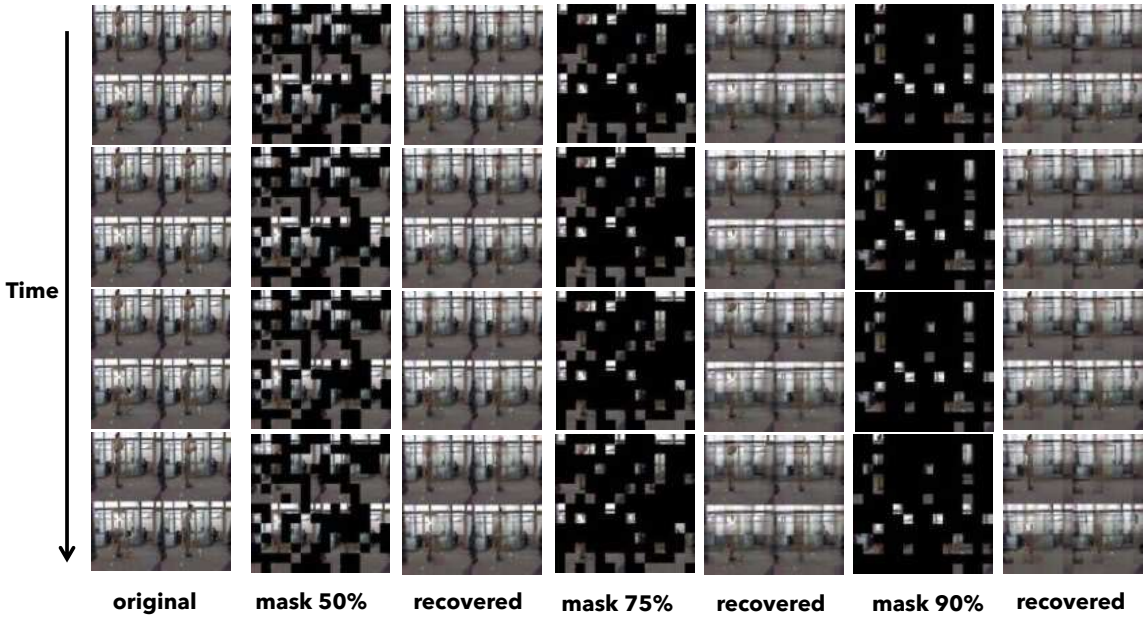


Figure 29. We present our new video sequence ($N = 2$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *put box on the floor* from the 3D Action Pairs RGB dataset.

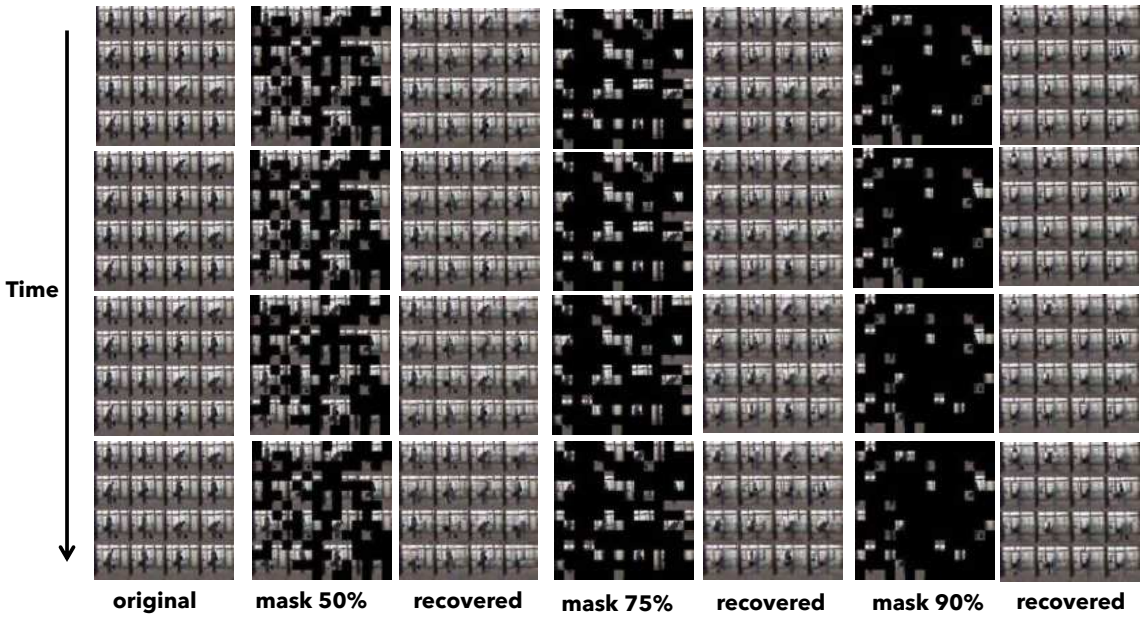


Figure 30. We present our new video sequence ($N = 4$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pick up bag and put it on back* from the 3D Action Pairs RGB dataset.

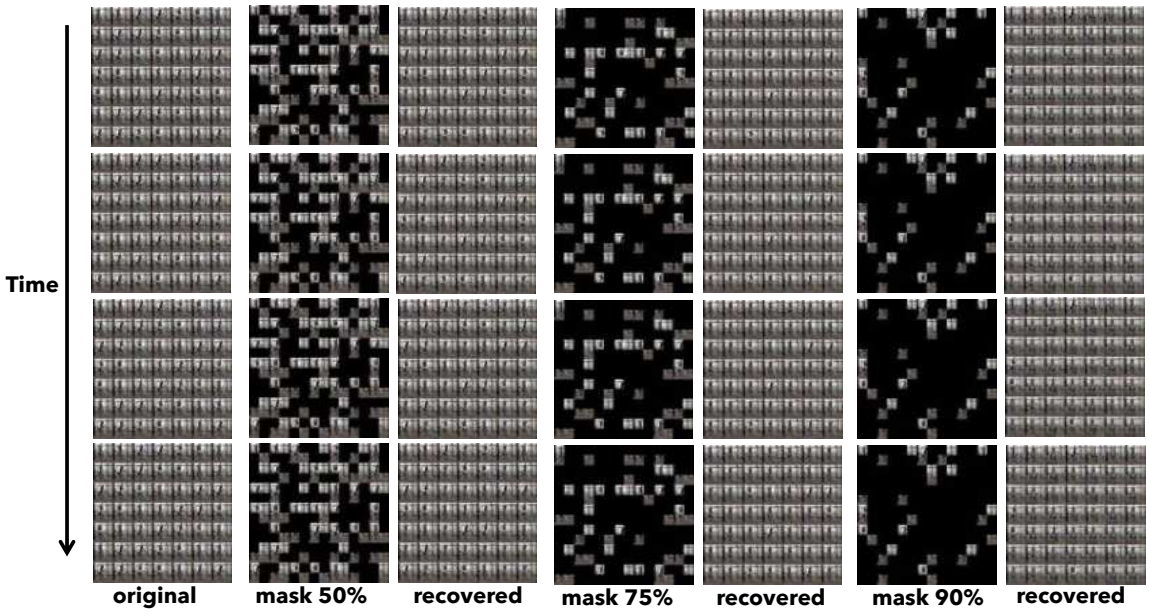


Figure 31. We present our new video sequence ($N = 7$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pick up hanging paper from desk side* from the 3D Action Pairs RGB dataset.

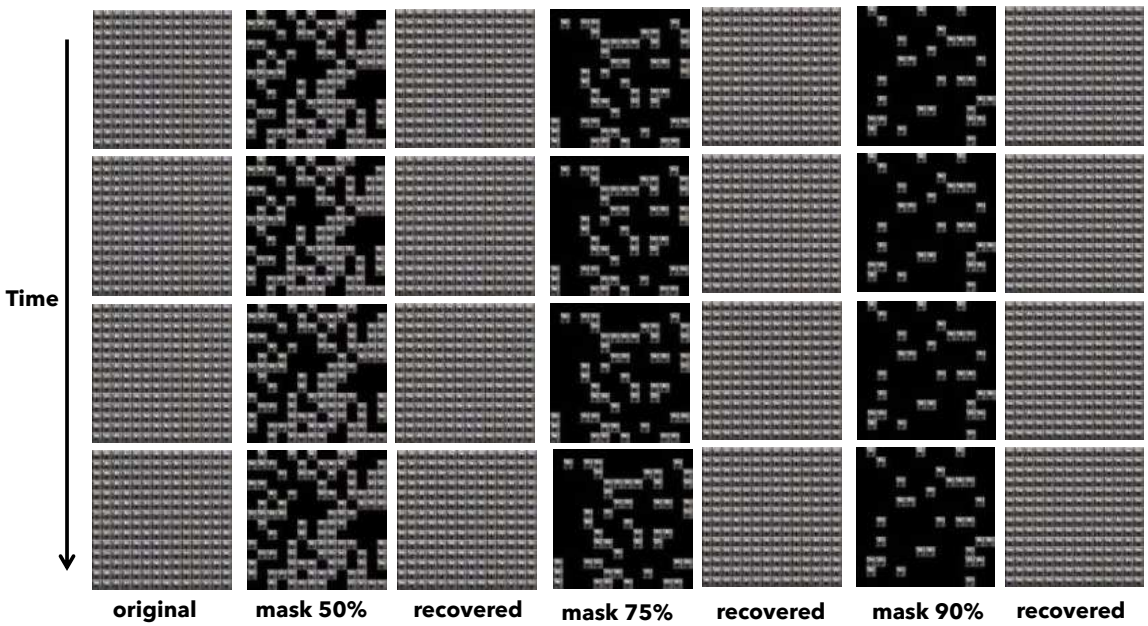


Figure 32. We present our new video sequence ($N = 14$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *put hat on head* from the 3D Action Pairs RGB dataset.

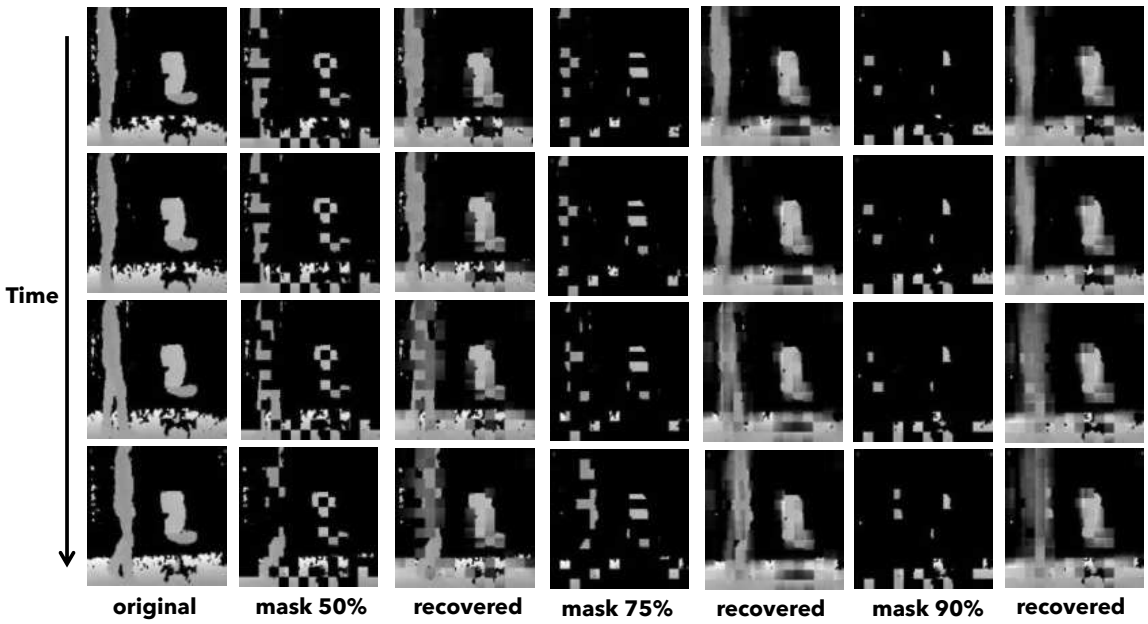


Figure 33. We present our new video sequence ($N = 1$, Baseline) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pull up the chair* from the 3D Action Pairs Depth dataset.

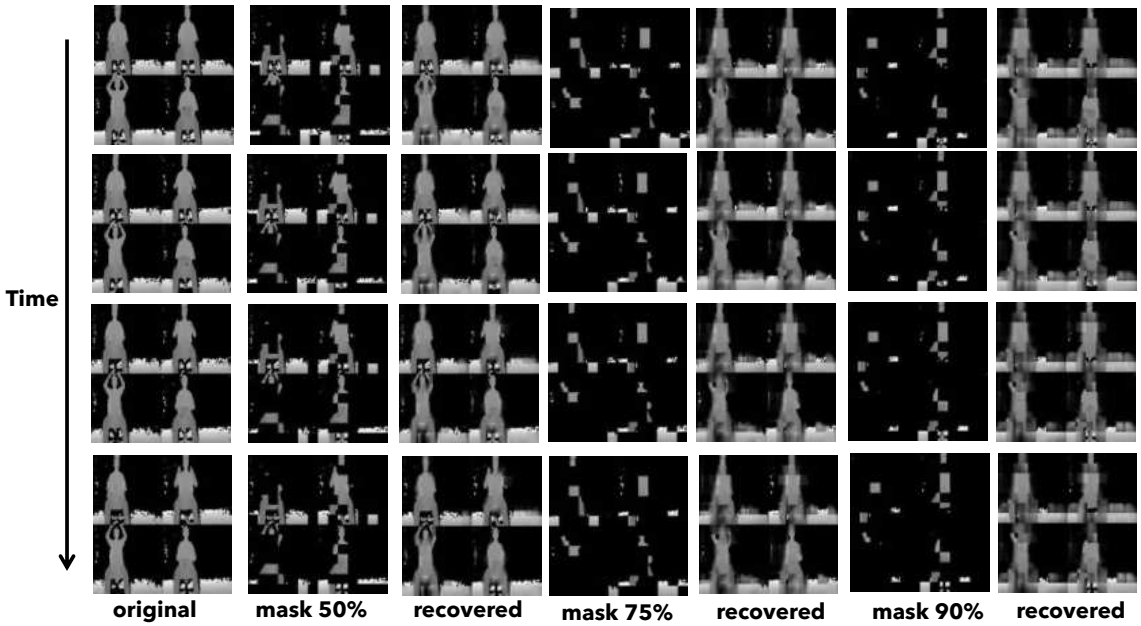


Figure 34. We present our new video sequence ($N = 2$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *take off the hat* from the 3D Action Pairs Depth dataset.

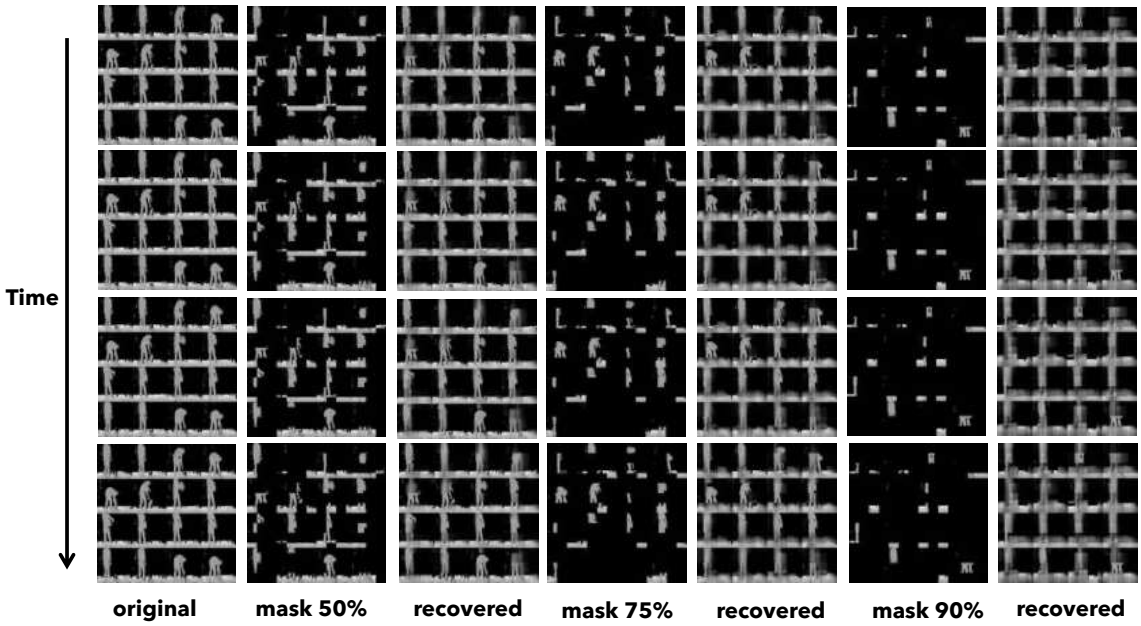


Figure 35. We present our new video sequence ($N = 4$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pick up the bag and put it on back* from the 3D Action Pairs Depth dataset.

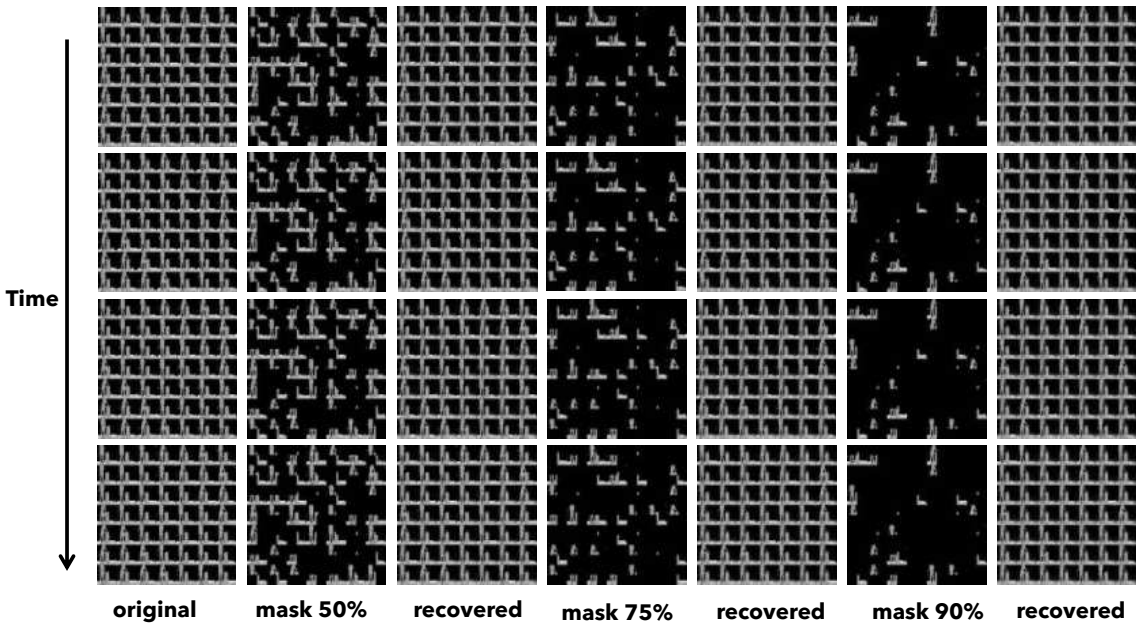


Figure 36. We present our new video sequence ($N = 7$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pick up hanging paper from desk side* from the 3D Action Pairs Depth dataset.

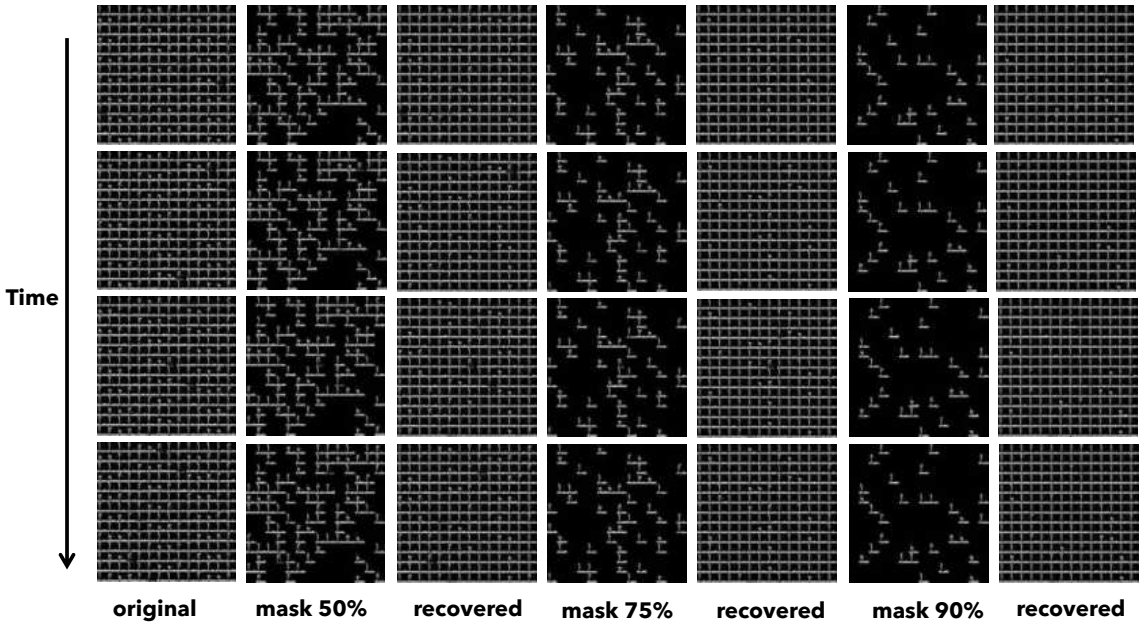


Figure 37. We present our new video sequence ($N = 14$) along with reconstructions at different masking ratios. The video reconstructions are predicted by VideoMAE with the TIME layer, pre-trained using masking ratios of 50%, 75%, and 90%. For this visualization, we select the action *pick up box on the floor* from the 3D Action Pairs Depth dataset.