MV2MAE: Self-Supervised Video Pre-Training with Motion-Aware Multi-View Masked Autoencoders

Anonymous authors
Paper under double-blind review

Abstract

Videos captured from multiple viewpoints can help in perceiving the 3D structure of the world and benefit computer vision tasks such as action recognition, tracking, etc. In this paper, we present MV2MAE, a method for self-supervised learning from synchronized multi-view videos, built on the masked autoencoder framework. We introduce two key enhancements to better exploit multi-view video data. First, we design a cross-view reconstruction task that leverages a cross-attention-based decoder to reconstruct a target viewpoint video from source view. This helps in effectively injecting geometric information and yielding representations robust to viewpoint changes. Second, we introduce a controllable motion-weighted reconstruction loss which emphasizes dynamic regions and mitigates trivial reconstruction of static backgrounds. This improves temporal modeling and encourages learning more meaningful representations across views. MV2MAE achieves state-of-the-art results on the NTU-60, NTU-120 and ETRI datasets among self-supervised approaches. In the more practical transfer learning setting, it delivers consistent gains of $+2.0 - 8.5\%$ on NUCLA, PKU-MMD-II and ROCOG-v2 datasets, demonstrating the robustness and generalizability of our approach. Code will be made publicly available.

1 Introduction

Multiple viewpoints of the same event are crucial to its understanding. Humans move around and obtain different viewpoints of objects and scenes, and develop a representation robust to viewpoint changes (Isik et al., 2018). Different viewpoints often have very different appearance, which can help address challenges due to occlusion, lighting variations and limited field-of-view. In many real world scenarios, we have videos captured from multiple viewpoints, *e.g.* sports videos (Saito et al., 2004), elderly care (Jang et al., 2020), self-driving (Yogamani et al., 2019), complex robotic manipulation tasks (Seo et al., 2023) and security videos (Corona et al., 2021). Learning a robust pre-trained model from large amounts of unlabeled synchronised multi-view data is of significant value for these applications. Such a model which is aware of the 3D geometry will be robust to changes in viewpoint and can be effectively used as a foundation for downstream finetuning on smaller datasets for different tasks.

There has been significant progress in video self-supervised learning (Schiappa et al., 2023) (SSL) for the single-view case, *i.e.* where synchronized multi-view data is not available. Recently, Masked Autoencoders (MAEs) as a paradigm for self-supervised learning has seen growing interest, and it has been successfully extended to video domain (Feichtenhofer et al., 2022; Tong et al., 2022; Wang et al., 2023a). MAE-based methods achieve superior performance (Tong et al., 2022) on standard datasets such as Kinetics-400 (Kay et al., 2017) and Something-Something-v2 (Goyal et al., 2017), compared to contrastive learning methods (Feichtenhofer et al., 2021). However, existing MAE-based pre-training approaches are not explicitly designed to be robust to viewpoint changes. View-invariant learning from multi-view videos has been widely studied using NTU (Shahroudy et al., 2016; Liu et al., 2020a) and ETRI (Jang et al., 2020) datasets. However, most of these methods use 3D human pose, which is difficult to accurately capture for in-the-wild scenarios. Recently, there has been a growing interest in RGB-based self-supervised learning approaches leveraging multi-view videos (Parameswaran & Chellappa, 2006; Das & Ryoo, 2023; Vyas et al., 2020; Li et al., 2018), facilitated by the availability of large-scale multi-view datasets (Shahroudy et al., 2016; Liu

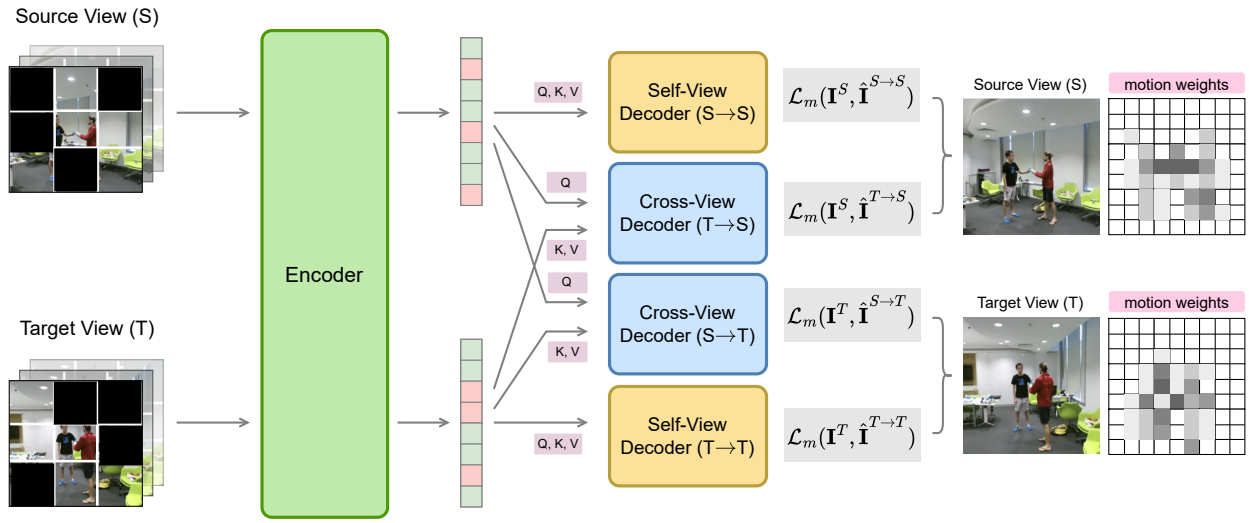

Figure 1: **Multi-View Video Masked Autoencoder (MV2MAE).** Given synchronized videos from a source (S) and target (T) viewpoint, we first tokenize and encode visible patches using a shared encoder. We introduce a cross-view decoder to reconstruct one viewpoint from the other, while the self-view decoder reconstructs each view independently. The proposed motion-weighted reconstruction loss $\mathcal{L}_m(\mathbf{I}, \hat{\mathbf{I}})$, computed using the input clip $\mathbf{I}$ and reconstructed clip $\hat{\mathbf{I}}$, emphasizes dynamic regions and reduces the effect of trivial reconstruction of static background regions. The motion weights are derived from the input clip.

et al., 2020a; Jang et al., 2020). ViewCLR (Das & Ryoo, 2023), which achieves state-of-the-art results among SSL methods, introduces a latent viewpoint generator as a learnable augmentation for generating positives in a contrastive learning (Chen et al., 2020) framework. However, this method is memory intensive as it requires storing two copies of the feature extractor and two queues of features, while also requiring multi-stage training. In contrast, the recent success of MAEs for video SSL motivates us to explore its potential in the *multi-view* video SSL scenario.

In this paper, we aim to learn self-supervised video representations that are robust to viewpoint shifts. Humans learn a representation robust to viewpoint variations for tasks such as action recognition and are able to *visualize how an action looks from different viewpoints* (Isik et al., 2018). Motivated by this, we design the task of using one viewpoint to predict the appearance from a different viewpoint, and integrate it in the MAE framework. More specifically, given a video of an activity from one viewpoint, it is patchified and a high fraction of the patches are masked out. The visible patches are encoded, which the decoder uses (along with `MASK` tokens for missing patches) to reconstruct the given video. We introduce an additional cross-view decoder, which is tasked with reconstructing the masked patches of a target viewpoint by using the visible regions from source view. This requires the model to understand the geometric relations between different views, making the pre-trained model robust to viewpoint shifts. Another challenge with MAE in videos is temporal redundancy, which makes it easier to reconstruct the static, background regions by simply copy pasting from adjacent frames where those are visible. Existing solutions for this problem involve specialized masking strategies using extra learnable modules (Bandara et al., 2023; Huang et al., 2023) or tube masking (Tong et al., 2022; Wang et al., 2023a), which are not effective in certain scenarios, *e.g.* when motion is localized in a small region of the frame. We propose a simple solution, without introducing additional learnable parameters, by modifying the reconstruction loss to focus on moving regions. We can control the relative weights of moving and static regions using a temperature parameter.

We perform pre-training experiments on three multi-view video datasets: NTU-60 (Shahroudy et al., 2016), NTU-120 (Liu et al., 2020a), ETRI (Jang et al., 2020). Our method achieves SOTA accuracy on these action recognition benchmarks in the full fine-tuning protocol. More notably, the robustness of our representation is shown in the transfer learning results on smaller datasets. We achieve SOTA results on NUCLA (Wang

et al., 2014), ROCOG-v2 (Reddy et al., 2023) and PKU-MMD-II (Liu et al., 2017) datasets in the transfer learning setting.

Our main contributions can be summarized as follows:

- We present MV2MAE, a self-supervised pre-training approach explicitly designed for synchronized multi-view videos, achieving state-of-the-art results among SSL methods on NTU-60, NTU-120, and ETRI benchmarks, as well as strong transfer gains (+2.0–8.5%) on NUCLA, PKU-MMD-II, and ROCOG-v2. We also show synthetic multi-view data as a scalable alternative for pre-training data.
- We introduce a dedicated cross-view decoder that reconstructs target viewpoint video from source view, forcing the encoder to capture 3D geometry and learn viewpoint-robust representations.
- We design a simple yet effective motion-aware loss that emphasizes moving regions while down-weighting static backgrounds, mitigating trivial reconstruction and substantially improving temporal representation learning.

2 Related Work

2.1 Self-Supervised Learning from Videos

Pretext Learning. Many pretext tasks have been proposed for learning self-supervised video representations, initially inspired from the progress in SSL for images. Tasks such as video rotation prediction (Jing et al., 2018), solving spatio-temporal jigsaw (Ahsan et al., 2019), predicting motion and appearance statistics (Wang et al., 2019) were direct extensions of their image counterparts, and showed impressive performance. Methods leveraging the temporal order in videos for constructing pretext tasks such as frame ordering (Xu et al., 2019) and odd-one-out learning (Fernando et al., 2017) were also proposed. These methods were outperformed by contrastive learning approaches.

Contrastive Learning. These methods create augmented versions of the input (positives) which preserve the semantic content of the input. The contrastive loss is used to pull these closer together in the feature space, while simultaneously pushing them away from other samples (negatives). Numerous ways of generating positive pairs were proposed such as using random clips from the same video, clips of different frame rates (Wang et al., 2020), choosing nearby clips (Qian et al., 2021), and using optical flow (Han et al., 2020), among others.

Masked Video Modeling. Recently, masked video modeling has emerged as a promising area for SSL. Methods such as BEVT (Wang et al., 2022), MaskFeat (Wei et al., 2022), VideoMAE (Tong et al., 2022), MAE-ST (Feichtenhofer et al., 2022) show superior performance on the standard video self-supervised learning benchmarks. Different reconstruction targets have been studied, such as MVD (Wang et al., 2023b) which uses distillation from pre-trained features, and MME (Sun et al., 2023) which reconstructs motion trajectories. To tackle trivial reconstruction solution via copy-paste in videos, which becomes an issue due to high redundancy, different masking strategies have been proposed. MGMAE (Huang et al., 2023) uses motion-guided masking based on motion vectors, VideoMAE (Tong et al., 2022) proposed using tube masking, AdaMAE (Bandara et al., 2023) introduces a neural network for mask sampling. Orthogonal to these, we propose to tackle the issue by using a motion-weighted reconstruction loss. Moreover, unlike the proposed approach, existing MAE pre-training approaches are not explicitly designed to be robust to viewpoint shifts.

2.2 Multi-View Action Recognition

Early works in this area designed hand-crafted features which are robust to viewpoint shifts (Parameswaran & Chellappa, 2006; Rao et al., 2002; Xia et al., 2012). Many unsupervised learning approaches have been proposed for learning representations robust to changes in viewpoint. A large number of methods leverage 3D human pose information, which greatly aids in achieving robustness to viewpoint variations. Methods based on RGB modality (Das & Ryoo, 2023; Li et al., 2018; Vyas et al., 2020) have gained increasing popularity. These can be broadly divided into two categories:

One trend is to enforce the latent representations of different viewpoints to be close. Along this line, (Zheng et al., 2012) follows a dictionary learning approach and encourages videos of different views to have the same sparse representation. (Rahmani & Mian, 2015) fits a 3D human model to a mocap sequence and generates videos from multiple viewpoints, which are forced to predict the same label. More recently, methods based on contrastive learning have been proposed such as ViewCLR (Das & Ryoo, 2023) which achieves remarkable performance. They add a latent viewpoint generator module which is used to generate positives in the latent space corresponding to different views.

Another line of work uses one viewpoint to predict another. (Li et al., 2018) uses cross-view prediction in 3D flow space by using depth as an additional input to provide view information. Their approach also uses a gradient reversal layer for achieving robustness to view changes. (Vyas et al., 2020) uses the encoded source view features to render same video from unseen viewpoint and a random start time. Their approach hence needs to be able to predict across time and viewpoint shifts. They leverage a view embedding which requires information of camera height, distance and angle. In contrast to these approaches which rely on view embedding or depth for providing viewpoint information, the view information is inherently available in the visible patches of the source and target viewpoints in our approach.

3 Method

3.1 Preliminary: Masked Video Modeling

Here we revisit the MAE framework for videos. Given a video, we first sample T frames with stride τ to get the input clip: $\mathbf{I} \in \mathbb{R}^{C \times T \times H \times W}$. Here, $H \times W$ is the spatial resolution, T denotes the number of frames sampled, and C is the number of input (RGB) channels. The standard MAE architecture has three main components: tokenizer, encoder & decoder.

Tokenizer. The input clip is first converted into N patches using a patch size of $t \times h \times w$, where $N = \frac{T}{t} \times \frac{H}{h} \times \frac{W}{w}$. The tokenizer returns N tokens of dimension d by first linearly embedding these N patches. This is implemented in practice using a strided 3D convolution layer. Next, we provide position information to these tokens by adding positional embeddings (Vaswani et al., 2017).

Encoder. A high fraction of these N tokens are dropped with a masking ratio $\rho \in (0, 1)$. Different masking strategies (Tong et al., 2022; Bandara et al., 2023; Huang et al., 2023) have been explored for choosing which tokens to mask out. Next, the remaining small fraction of visible tokens are passed through the encoder ($\Phi_{\texttt{enc}}$) to obtain latent representations. The encoder is a vanilla ViT (Dosovitskiy et al., 2020) with joint space-time attention (Tong et al., 2022). These latent representations need to capture the semantics in order to reconstruct the masked patches.

Decoder. The encoded latent representations of the visible patches are concatenated with learnable `[MASK]` tokens corresponding to masked-out patches, resulting in combined tokens $\mathbf{Z}$. The positional embeddings are then added for all tokens, and passed through a light-weight decoder ($\Phi_{\texttt{dec}}$) to get the predicted pixel values $\hat{\mathbf{I}} = \Phi_{\texttt{dec}}(\mathbf{Z})$.

The loss function is the mean squared error (MSE) between the reconstructed values and the normalized pixel values (Feichtenhofer et al., 2022; Tong et al., 2022), for masked patches Ω.

$$\mathcal{L}(\mathbf{I}, \hat{\mathbf{I}}) = \frac{1}{\rho N} \sum_{i \in \Omega} |\mathbf{I}_i - \hat{\mathbf{I}}_i|^2 \tag{1}$$

3.2 Cross-View Reconstruction

The goal of cross-view reconstruction is to predict the missing appearance of a video in target viewpoint using the visible patches of video from source viewpoint. Being able to extrapolate across viewpoints requires understanding the geometric relations between different viewpoints, making it an effective task for learning representations robust to viewpoint variations.

As shown in Figure 1, consider two synchronized videos of an activity, $\mathbf{I}^S$ and $\mathbf{I}^T$, from source view (S) and target view (T) respectively. We first tokenize, mask and encode the visible tokens for each video separately using a shared encoder $\Phi_{\texttt{enc}}$. We then introduce a cross-view decoder ($\Phi_{\texttt{dec}}^{\texttt{cross-view}}$) which additionally uses the visible tokens in the source view to reconstruct the target viewpoint video, $\hat{\mathbf{I}}^{S\rightarrow T} = \Phi_{\texttt{dec}}^{\texttt{cross-view}}(\mathbf{Z}^T, \mathbf{Z}^S_{\texttt{vis}})$. More specifically, each block of the cross-view decoder consists of cross-attention and self-attention layers, followed by a feed-forward layer. The tokens from the target view attend to the visible source view tokens using cross-attention, and then to each other using self-attention. Moreover, the standard decoder ($\Phi_{\texttt{dec}}$) is used to reconstruct video from each viewpoint independently $\hat{\mathbf{I}}^{v\rightarrow v} = \Phi_{\texttt{dec}}(\mathbf{Z}^v)$ for $v \in \{S, T\}$. Figure 5 visualizes the cross-view reconstruction quality and cross-attention maps, which demonstrates that the model learns to focus on relevant regions across viewpoints.

A key aspect of methods using the cross-view prediction paradigm is how the viewpoint information is provided: (Vyas et al., 2020) conditions the decoder on a viewpoint embedding, while some approaches (Li et al., 2018) use extra modalities such as depth to provide information about the target viewpoint. In contrast to these, in our approach, the visible patches provide the required target viewpoint information. The *amount* of view information we want to provide can be easily varied by changing the masking ratio.

3.3 Motion-Weighted Reconstruction Loss

A given video can be decomposed into static and dynamic regions. Static regions typically involve scene background and objects which do not move throughout the video. Patches from such regions are trivial to reconstruct (Sun et al., 2023; Bandara et al., 2023) due to temporal redundancy in videos. In order to deal with this, we offer a simple solution by re-weighting the reconstruction loss of each patch proportional to the amount of motion within that patch. The motion weights used for re-weighting are obtained using frame difference for simplicity. Note that other motion features such as optical flow, motion history image, etc can be used in place of frame difference, but frame difference is extremely fast to compute. In order to get the final weights, we take the norm of frame difference within each patch, and apply temperature softmax over all tokens. We can control the extent to which to focus on the moving regions by controlling the temperature parameter. The higher the temperature value, the more uniform the resulting weights. Examples of motion weights overlaid on the original frames for different temperature values are shown in Figure 2. PyTorch-style code for computing the motion weights for a clip is provided in the supplementary. The final motion-weighted reconstruction loss ($\mathcal{L}_{\text{m}}$) is given below, where $w_i(\tau)$ is the weight for i^{th} patch with temperature τ:

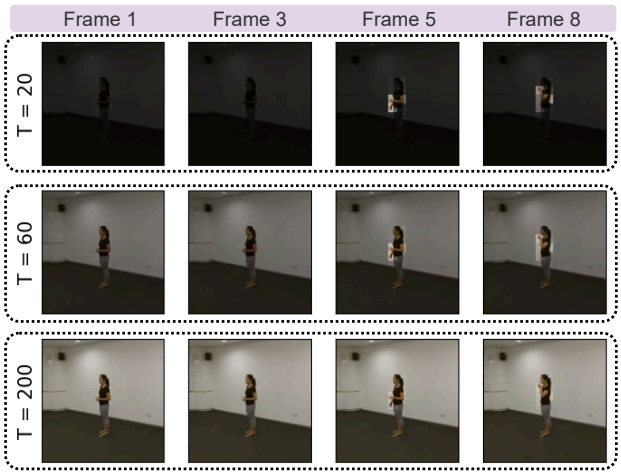

Figure 2: **Motion weights with varying temperature.** Each row shows motion weights overlaid on the input frames for a different temperature. Higher temperature increases the weight on static and background regions.

$$\mathcal{L}_{\text{m}}(\mathbf{I}, \hat{\mathbf{I}}; \tau) = \frac{1}{\rho N} \sum_{i \in \Omega} w_i(\tau) \times |\mathbf{I}_i - \hat{\mathbf{I}}_i|^2 \tag{2}$$

3.4 Pre-training and Evaluation

Pre-training. To sum up, we first sample two videos from a set of synchronized videos of an activity, $\mathbf{I}^S$ from source view and $\mathbf{I}^T$ from target view. The encoder, self-view decoder and cross-view decoder are pre-trained using the overall loss given below:

$$\mathcal{L} = \underbrace{\mathcal{L}_{\mathrm{m}}(\mathbf{I}^S, \hat{\mathbf{I}}^{S\to S}) + \mathcal{L}_{\mathrm{m}}(\mathbf{I}^T, \hat{\mathbf{I}}^{T\to T})}_{\text{self-view reconstruction}} + \underbrace{\lambda\mathcal{L}_{\mathrm{m}}(\mathbf{I}^T, \hat{\mathbf{I}}^{S\to T}) + \lambda\mathcal{L}_{\mathrm{m}}(\mathbf{I}^S, \hat{\mathbf{I}}^{T\to S})}_{\text{cross-view reconstruction}} \tag{3}$$

Here, $\hat{\mathbf{I}}^{S\to T}$ denotes reconstructed video from target view using source view, and so on. λ is the weight for cross-view reconstruction loss and is set to 1.

Evaluation. Following prior work (Li et al., 2018; Kim et al., 2022; Vyas et al., 2020; Das & Ryoo, 2023), we evaluate our pre-trained models on the task of action recognition using two settings: 1) full fine-tuning on the same datasets and 2) transfer learning on smaller datasets. We discard the decoders, and attach a classifier head which uses the global average pooled features for classification. For testing, we sample 5 temporal clips, and use 10 crops from each following (Das & Ryoo, 2023), taking their average for the final prediction.

4 Experiments

We evaluate our approach on several common multi-view video datasets: NTU60 (Shahroudy et al., 2016), NTU120 (Liu et al., 2020a), ETRI (Jang et al., 2020), NUCLA (Wang et al., 2014), PKU-MMD (Liu et al., 2017), and ROCOG (Reddy et al., 2023). For NTU and ETRI, we achieve state-of-the-art results among SSL methods by pre-training and fine-tuning on the target domain. On NUCLA, PKU-MMD, and ROCOG, we demonstrate excellent transfer learning performance by pre-training only on NTU, and fine-tuning on the target dataset.

4.1 Datasets

NTU RGB+D 60. (Shahroudy et al., 2016) is a large-scale multi-view action recognition dataset, consisting of 56,880 videos from 60 distinct action classes. These videos were recorded from 40 subjects using Kinect-v2. Each activity instance is simultaneously captured from three viewpoints. The dataset consists of two benchmarks outlined in (Shahroudy et al., 2016): (1) Cross-Subject (xsub) and (2) Cross-View (xview). In the cross-subject benchmark, the 40 subjects are divided into training and testing sets, with 20 subjects in each. In the cross-view scenario, videos from cameras 2 and 3 are used for training, while testing is performed on videos from camera 1.

NTU RGB+D 120. (Liu et al., 2020a) is an extended version of the NTU-60 dataset containing 114,480 videos spanning 120 action categories. Our evaluation follows the established protocols outlined in (Liu et al., 2020a): (1) Cross-Subject (xsub) and (2) Cross-Setup (xset). In the cross-subject scenario, subjects are partitioned into training and testing groups, while in the cross-setup setting, the data is divided into training and testing subsets based on the setup ID.

ETRI. (Jang et al., 2020) is another large-scale multi-view action recognition dataset consisting of activities of daily living for elderly care. It has 112,620 videos captured from 55 action classes. All activity instances are recorded from 8 synchronized viewpoints. (Jang et al., 2020) describes a cross-subject benchmark which we use to evaluate our approach.

4.2 Implementation Details

We sample a clip of 16 RGB frames with a stride of 4 from each video. We downsample the resolution of frames to 128×128 following (Das & Ryoo, 2023). During pre-training, we only apply random resized crops as augmentation. We use a temporal patch size of 2 and a spatial patch size of 16×16, which results in 512 tokens. A masking ratio of 0.7 is used unless otherwise specified. We choose fixed sinusoidal spatio-temporal positional embedding following (Tong et al., 2022; Bandara et al., 2023). All of our experiments use the vanilla ViT-S/16 (Touvron et al., 2021) architecture as the encoder, trained using AdamW optimizer (Loshchilov & Hutter, 2017). The pre-training is carried out for 1600 epochs. Please refer to the supplementary material for additional details.

Method	Modality	Resolution	# Frames	NTU-60 (%)	
				xview	xsub
Supervised Methods					
STA-Hands (Baradel et al., 2017)	RGB+Pose	299×299	20	88.6	82.5
Separable STA (Das et al., 2019)	RGB+Pose	–	64	94.6	92.2
ESE-FN (Shu et al., 2022)	RGB+Pose	–	64	96.7	92.4
VPN (Das et al., 2020)	RGB+Pose	–	64	96.2	93.5
VPN++ (Das et al., 2021)	RGB+Pose	–	64	99.1	96.6
3DA (Kim et al., 2023)	RGB+Pose	224×224	12	97.9	94.3
PoseC3D (Duan et al., 2022)	RGB+Pose	–	32	99.6	97.0
DA-Net (Wang et al., 2018)	RGB	–	–	75.3	–
Zhang *et al.* (Zhang et al., 2018)	RGB	–	–	70.6	63.3
Glimpse Clouds (Baradel et al., 2018)	RGB	224×224	8	93.2	86.6
DMCL (Garcia et al., 2021)	RGB	224×224	8	–	83.6
Debnath *et al.* (Debnath et al., 2021)	RGB	–	–	–	87.2
FSA-CNN (Jang et al., 2020)	RGB	–	–	92.2	88.1
Piergiovanni *et al.* (Piergiovanni & Ryoo, 2021)	RGB	–	–	93.7	–
ViewCon (Shah et al., 2023b)	RGB	224×224	32	98.0	91.4
DVANet (Siddiqui et al., 2024)	RGB	–	–	98.2	93.4
π-ViT (Reilly & Das, 2024)	RGB	224×224	16	97.9	94.0
Self-Supervised Methods					
Li *et al.* (Li et al., 2018)	Flow	–	6	83.4	80.9
GL-Transformer (Kim et al., 2022)	Pose	–	300	83.8	76.3
AimCLR (Guo et al., 2022)	Pose	–	–	89.2	83.0
HaLP (Shah et al., 2023a)	Pose	–	64	88.6	82.1
Vyas *et al.* (Vyas et al., 2020)	RGB	112×112	–	86.3	82.3
ViewCLR (Das & Ryoo, 2023)	RGB	128×128	32	94.1	89.7
MV2MAE (Ours)	RGB	128×128	16	**95.9**	**90.0**

Table 1: Comparison with state-of-the-art on cross-view and cross-subject action recognition benchmarks of NTU-60 dataset. **Top:** supervised methods using multiple modalities (RGB+Pose), and RGB only, **Bottom:** self-supervised methods using any modality. We report top-1 accuracy after finetuning as in (Li et al., 2018; Das & Ryoo, 2023; Vyas et al., 2020; Kim et al., 2022)

4.3 Comparison with state-of-the-art

We compare our approach with prior supervised and self-supervised approaches on the cross-subject (xsub) and cross-view (xview) benchmarks of the commonly used NTU-60 and NTU-120 datasets. We also present our results on the ETRI dataset, which only has a cross-subject benchmark.

Table 1 and Table 2 show results on the NTU-60 and NTU-120 datasets. We outperform all previous *self-supervised methods* based on RGB, Flow or Pose modality on both cross-view and cross-subject benchmarks of the two datasets. In the xsub setting, we see an improvement of +0.3% and +1.2% on NTU-60 and NTU-120 respectively, and in the xview setting, we observe an improvement of +1.8% and +0.9% respectively. Our approach is also faster to train (Tong et al., 2022) and more memory efficient compared to ViewCLR (Das & Ryoo, 2023), which uses a MoCo (Chen et al., 2020) framework and hence requires storing two copies of the model and two queues in memory. Vyas *et al.* (Vyas et al., 2020) uses a cross-view prediction paradigm but underperforms (86.3% vs 95.9% on xview and 82.3% vs 90.0% on NTU60 xsub) despite using more parameters (∼72M vs ∼22M). Unlike their approach which relies only on learnt viewpoint embeddings for information of the target viewpoint, we implicitly use the view information from the visible patches of target viewpoint. Moreover, it is noteworthy that most of the *supervised methods* (in Table 1, 2, 4, 6) use a resolution of 224×224 or higher, and despite using a much lower resolution of 128×128, MV2MAE shows strong performance.

Method	Modality	Resolution	# Frames	**NTU-120 (%)** xset	xsub
Supervised Methods					
Hu *et al.* (Hu et al., 2018)	RGB+Depth	–	–	44.9	36.3
Hu *et al.* (Hu et al., 2015)	RGB+Depth	–	–	54.7	50.8
Separable STA (Das et al., 2019)	RGB+Pose	–	64	82.5	83.8
VPN (Das et al., 2020)	RGB+Pose	–	64	87.8	86.3
VPN++ (Das et al., 2021)	RGB+Pose	–	64	92.5	90.7
3DA (Kim et al., 2023)	RGB+Pose	224×224	12	91.4	90.5
PoseC3D (Duan et al., 2022)	RGB+Pose	–	32	96.4	95.3
PEM (Liu & Yuan, 2018)	Pose	–	–	66.9	64.6
2s-AGCN (Lei et al., 2019)	Pose	–	300	84.9	82.9
MS-G3D Net (Liu et al., 2020b)	Pose	–	300	88.4	86.9
CTR-GCN (Chen et al., 2021)	Pose	–	64	90.6	88.9
Two-streams (Simonyan & Zisserman, 2014)	RGB	–	–	54.8	58.5
Liu *et al.* (Liu et al., 2020a)	RGB	–	*all*	54.8	58.5
I3D (Carreira & Zisserman, 2017)	RGB	224×224	250	80.1	77.0
DMCL (Garcia et al., 2021)	RGB	224×224	8	84.3	–
ViewCon (Shah et al., 2023b)	RGB	224×224	32	87.5	85.6
DVANet (Siddiqui et al., 2024)	RGB	–	–	91.6	90.4
π-ViT (Reilly & Das, 2024)	RGB	224×224	16	92.9	91.9
Self-Supervised Methods					
GL-Transformer (Kim et al., 2022)	Pose	–	300	68.7	66.0
AimCLR (Guo et al., 2022)	Pose	–	–	76.7	76.4
HaLP (Shah et al., 2023a)	Pose	–	64	73.1	72.6
ViewCLR (Das & Ryoo, 2023)	RGB	128×128	32	86.2	84.5
MV2MAE (Ours)	RGB	128×128	16	**87.1**	**85.3**

Table 2: Comparison with state-of-the-art on cross-setup and cross-subject action recognition benchmarks of NTU-120 dataset. **Top:** supervised methods using multiple or single modality, **Bottom:** self-supervised methods using any modality. We report top-1 accuracy after finetuning as in (Das & Ryoo, 2023; Vyas et al., 2020; Kim et al., 2022), *etc.*

Method	Modality	Resolution	# Frames	**ETRI (%)** xsub
Supervised Methods				
ESE-FN (Shu et al., 2022)	RGB+Pose	–	64	95.9
FSA-CNN (Jang et al., 2020)	RGB	–	–	90.6
ConViViT (Dokkar et al., 2023)	RGB	–	–	95.1
Self-Supervised Methods				
MV2MAE (Ours)	RGB	128×128	16	**96.5**

Table 3: Comparison with state-of-the-art cross-subject action recognition benchmark of ETRI dataset. MV2MAE performs better than prior work, which are all supervised approaches.

On the ETRI dataset (Table 3), MV2MAE improves the action classification accuracy by +1.4% compared to (Dokkar et al., 2023).

4.4 Transfer Learning Results

Transfer learning is an important setting for evaluating the generalization capabilities of pre-trained models. The model is initialized using pre-trained weights, and fine-tuned on smaller datasets. We perform transfer learning experiments on three action recognition datasets: 1) NUCLA, 2) PKU-MMD-II, and 3) ROCOG-v2.

Method	Modality	Resolution	# Frames	**NUCLA (%)** xview
Self-Supervised Methods				
STA (Das et al., 2019)	RGB+Pose	–	64	92.4
VPN (Das et al., 2020)	RGB+Pose	–	64	93.5
VPN++ (Das et al., 2021)	RGB+Pose	–	64	93.5
DA-Net (Wang et al., 2018)	RGB	–	–	86.5
Glimpse Cloud (Baradel et al., 2018)	RGB	224×224	8	90.1
I3D (Carreira & Zisserman, 2017)	RGB	224×224	250	88.8
ViewCon (Shah et al., 2023b)	RGB	224×224	32	91.7
DVANet (Siddiqui et al., 2024)	RGB	–	–	96.5
Self-Supervised Methods				
MS^2L (Lin et al., 2020)	Pose	–	200	86.8
Li *et al.* (Li et al., 2018)	Depth	–	6	62.5
Colorization (Yang et al., 2021)	Depth	–	50	94.0
Vyas *et al.* (Vyas et al., 2020)	RGB	112×112	–	83.1
ViewCLR (Das & Ryoo, 2023)	RGB	128×128	32	89.1
MV2MAE (Ours)	RGB	128×128	16	**97.6**

Table 4: Transfer learning on NUCLA. Self-supervised methods are pre-trained on NTU-60 dataset. MV2MAE significantly outperforms other methods showing remarkable transfer capability of our representations.

Method	Modality	Resolution	# Frames	**PKU-MMD (%)** xsub
Self-Supervised Methods				
CrosSCLR-B (Zolfaghari et al., 2021)	Pose	–	–	52.8
CMD (Mao et al., 2022)	Pose	–	64	57.0
HaLP (Shah et al., 2023a)	Pose	–	64	57.3
MV2MAE (Ours)	RGB	128×128	16	**60.1**

Table 5: Transfer learning on PKU-MMD. All methods use NTU-120 dataset for pre-training. MV2MAE surpasses other unsupervised methods, all of which use Pose modality.

NUCLA (Wang et al., 2014) is a multi-view action recognition dataset consisting of 1493 videos spanning 10 action classes. Each activity has been captured from three viewpoints, and we follow the cross-view protocol for our experiments. PKU-MMD-II (Liu et al., 2017) is another dataset for 3D action understanding, consisting of 6945 videos from 51 activity classes. Following prior work (Shah et al., 2023a), we use the phase 2 of the dataset and evaluate our approach on the cross-subject setting. ROCOG-v2 (Reddy et al., 2023) is a gesture recognition dataset consisting of 304 ground viewpoint videos from 7 gestures.

As shown in Table 4, our method achieves significantly better performance than prior *supervised and unsupervised* methods on the NUCLA dataset. MV2MAE improves the action classification accuracy by +8.5% upon the previous RGB-based SOTA SSL approach (Das & Ryoo, 2023) and by +1.1% compared to best *supervised* pre-trained approach (Siddiqui et al., 2024). On the PKU-MMD-II dataset (Table 5), our method shows an improvement of +2.8% compared to prior work, all of which are based on Pose modality. Finally, we show results on the ROCOG-v2 dataset in Table 6, where we gain an improvement of +2.0%.

These transfer learning results clearly demonstrates that the representations learnt using our approach generalize well. It is noteworthy that although the Pose modality shows superior performance in supervised setting (Table 2), it lags behind when used for self-supervised learning in both in-domain fine-tuning (Table 2) and transfer learning (Table 4 and 5) settings.

Method	Modality	Resolution	# Frames	**ROCOG (%)**
Reddy *et al.* (Reddy et al., 2023)	RGB	256×256	16	87.0
MV2MAE (Ours)	RGB	128×128	16	**89.0**

Table 6: Transfer learning on ROCOG ground dataset.

4.5 Synthetic Multi-View Data for Pre-Training

Real multi-view videos can be difficult to acquire and can pose privacy concerns. As an alternative, we investigate the use of synthetic multi-view action recognition data. In our experiments, we pre-train models using synthetic data from SynADL (Hwang et al., 2021) dataset and fine-tune and evaluate on real data from ETRI (Jang et al., 2020) dataset. We compare synthetic pre-training (green) with real pre-training (orange). We observe that if the amount of synthetic data used is same (1x) as the amount of real data, there is a performance drop due to the domain difference. However, increasing the amount of synthetic data used for pre-training allows synthetic pre-training to surpass real pre-training, as seen in Figure 3. These results suggest that synthetic data can serve as a scalable and effective alternative for pre-training.

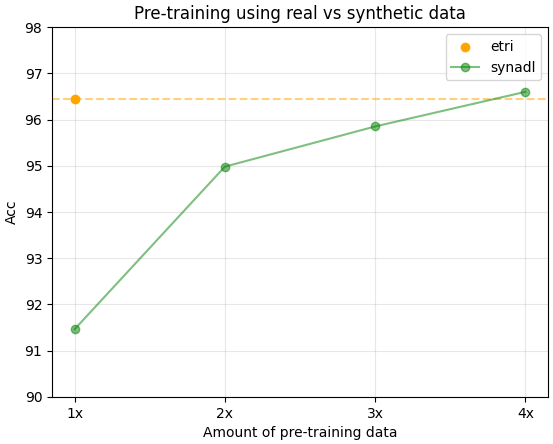

Figure 3: **Pre-training using synthetic data.** Pre-training using more (4x) synthetic data beats pre-training using real data on the same real test set.

4.6 Ablation Study and Analysis

We perform ablation experiments on the cross-subject benchmark of the NTU-120 dataset. For ablations regarding viewpoints (Table 10 and 7), we use the ETRI dataset as it contains four synchronised viewpoints compared to NTU-120 which has three.

How much emphasis to place on reconstructing moving patches? The motion weights in MV2MAE can be adjusted to modulate the emphasis on moving patches, using the temperature parameter. As shown in Figure 2, a lower temperature places more focus on reconstructing patches with more motion, and increasing the temperature increases the weight given to the background pixels. Figure 4 shows the influence of the temperature parameter on accuracy. From the plot, we see that a temperature value of 60 performs best, which is used in all our experiments. Increasing the weights of background patches by increasing temperature degrades the performance. This is because it is trivial to reconstruct the background patches by copy-paste from nearby frames. The performance degrades significantly to 82.46% if each patch in the video is weighted equally, since the the number of static pixels are more than those with motion.

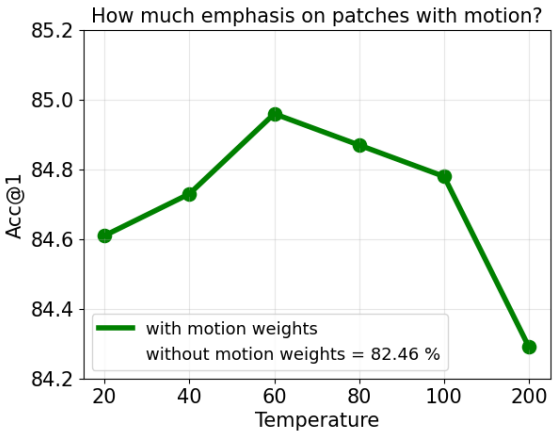

Figure 4: **Temperature parameter** τ of motion weights modulates the focus on static vs moving regions as visualized in Figure 2. Accuracy drops by

Masking Ratio. We study the impact of masking ratio in Table 7. Note that the optimal masking ratio is lower in our multi-view setting than the standard MAE (Tong et al., 2022), which we hypothesize is because more information is needed from individual views to effectively infer cross-view geometry.

Model Scaling. To study how the performance scales with models of different capacities, we pre-train using ViT-T, ViT-S, and ViT-B encoders and compare the fine-tuning accuracy on the NTU-120 cross-subject setting in Table 8. MV2MAE can effectively pre-train larger models using the same amount of data. All model scaling experiments perform pre-training for 800 epochs as opposed to the default 1600 due to computational constraints.

Masking ratio (ρ)	0.6	0.7	0.8	0.9
NTU-120 xsub (%)	83.6	**85.3**	84.3	83.4

Table 7: **Masking Ratio.** MV2MAE performs best with a masking ratio of 0.7. Default in gray.

Backbone	ViT-T	ViT-S	ViT-B
NTU-120 xsub (%)	82.0	83.4	85.1

Table 8: **Model Capacity.** Our approach scales effectively with bigger models. Default in gray.

Comparison with VideoMAE. On the NTU-120 xsub benchmark, VideoMAE (Tong et al., 2022) achieves 79.7% action recognition accuracy using its tube masking strategy, compared to MV2MAE which achieves 85.3% accuracy.

Ablation of Cross-view Decoder. The cross-view decoder significantly boosts performance in both xsub and xset settings, with more improvement on the xview benchmark (+2.9%) than on xsub benchmark (+2.2%), showing effectiveness when evaluating on unknown viewpoints.

Cross-View Decoder	NTU-120 xsub (%)	NTU-120 xset (%)
✗	83.1	84.2
✓	85.3 (+2.2)	87.1 (+2.9)

Table 9: **Cross-View Decoder** leads to substantial improvements on both xview and xset benchmarks.

Visualizing Cross-Attention Maps and Reconstructions. Here, we analyze the cross-view decoder by visualizing the cross-attention maps and the cross-view reconstruction quality. The cross-attention maps are visualized in Figure 5. The first and second rows show the input and masked input frames from the target viewpoint, with a masked query token circled in red. The third row shows the reconstructed target view from the cross-view decoder. The last row shows the cross-attention map corresponding to the query token overlaid on the source view frames. We can see that model is able to find matching regions in the source viewpoint, demonstrating the learnt geometry.

How many source views to use? For the cross-view decoder, we study the effect of number of source viewpoints used in Table 10. For these experiments, all viewpoints used are chosen randomly from available synchronized views. The performance is similar when using one or two source viewpoints. We observe that the fine-tuning accuracy drops if we use more source viewpoints for reconstructing the target viewpoint, by making the reconstruction task easier.

# Source Views	1	2	3
ETRI xsub (%)	94.0	93.9	93.1

Table 10: **Number of Source Views.** Using more source views makes reconstruction task easier and degrades performance. Default in gray.

How different should the views be? A natural question that arises when creating such datasets is which viewpoint to capture? Given a target viewpoint, we study how far should the source view be. We study this by fixing the target view to be View1 (shown in Figure 6), and varying the source view to be View2, View3 or View4 . The results are reported in Table 7, which shows that the performance drops slightly if the chosen target and source viewpoints are separated by a lot. For all experiments in the paper, we do not fix the target view, and choose source and target views randomly for added diversity.

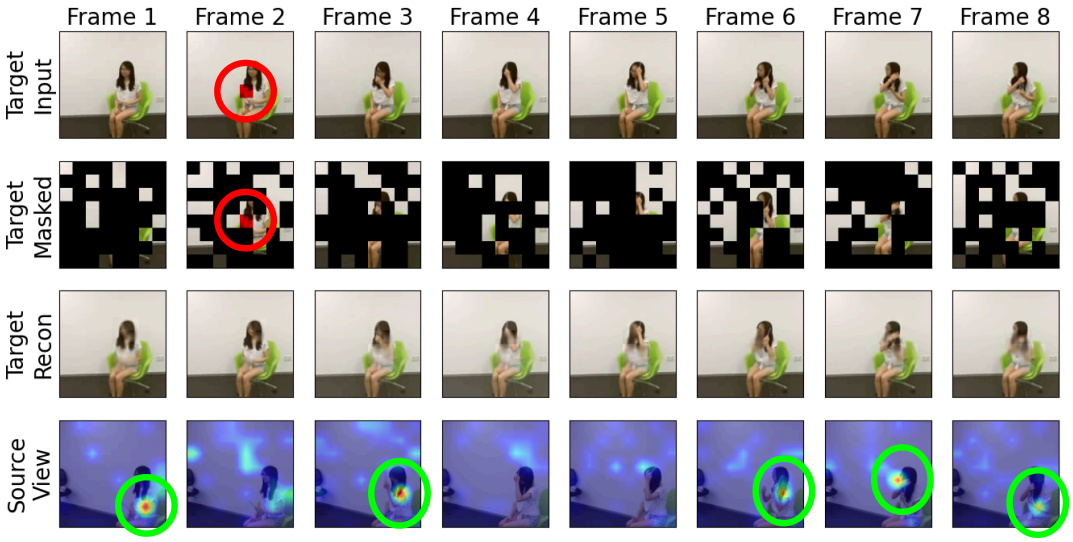

Figure 5: **Cross-View Decoder Qualitative Analysis.** We visualize the reconstructions and cross-attention maps from the cross-view decoder. **First row:** Target viewpoint input frames, **Second row:** Masked input frames from target viewpoint, **Third row:** Reconstruction of target view from the cross-view decoder, **Last row:** Cross-attention maps visualized on source view frames. The red circle indicates a query token in the target viewpoint whose attention maps are visualized in the last row. Green circles shows that the model is able find and leverage matching regions across viewpoints for cross-view reconstruction.

Figure 6: Example of synced views from ETRI dataset.

Source View	View2	View3	View4
ETRI xsub (%)	94.7	94.4	94.3

Figure 7: **Which views to choose?** View2 is the closest and View4 the farthest from the target. Performance degrades slightly if the two views are very different.

5 Conclusion

We propose a self-supervised learning approach for harnessing the power of multi-view videos within the masked autoencoder framework. Our method integrates a cross-view reconstruction task, leveraging a dedicated decoder equipped with cross-attention mechanism to instill geometry information into the model. The introduction of a motion-focused reconstruction loss further enhances temporal modeling. We empirically show that our SSL method enables learning robust generalizable multi-view features contributing to better performance when used for full fine-tuning and transfer learning.

References

Unaiza Ahsan, Rishi Madhok, and Irfan Essa. Video jigsaw: Unsupervised learning of spatiotemporal context for video action recognition. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 179–189. IEEE, 2019.

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*, pp. 6836–6846, 2021.

Wele Gedara Chaminda Bandara, Naman Patel, Ali Gholami, Mehdi Nikkhah, Motilal Agrawal, and Vishal M Patel. Adamae: Adaptive masking for efficient spatiotemporal learning with masked autoen-

coders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14507–14517, 2023.

Fabien Baradel, Christian Wolf, and Julien Mille. Human action recognition: Pose-based attention draws focus to hands. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 604–613, 2017.

Fabien Baradel, Christian Wolf, Julien Mille, and Graham W. Taylor. Glimpse clouds: Human activity recognition from unstructured feature points. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733. IEEE, 2017.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13359–13368, 2021.

Kellie Corona, Katie Osterdahl, Roderic Collins, and Anthony Hoogs. Meva: A large-scale multiview, multimodal video dataset for activity detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1060–1068, January 2021.

Srijan Das and Michael S Ryoo. Viewclr: Learning self-supervised video representation for unseen viewpoints. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 5573–5583, 2023.

Srijan Das, Rui Dai, Michal Koperski, Luca Minciullo, Lorenzo Garattoni, Francois Bremond, and Gianpiero Francesca. Toyota smarthome: Real-world activities of daily living. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Srijan Das, Saurav Sharma, Rui Dai, Francois Bremond, and Monique Thonnat. VPN: Learning video-pose embedding for activities of daily living. In *European Conference on Computer Vision*, pp. 72–90. Springer, 2020.

Srijan Das, Rui Dai, Di Yang, and Francois Bremond. VPN++: Rethinking video-pose embeddings for understanding activities of daily living. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9703–9717, 2021.

Bappaditya Debnath, Mary O'brien, Swagat Kumar, and Ardhendu Behera. Attentional learn-able pooling for human activity recognition. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13049–13055. IEEE, 2021.

Rachid Reda Dokkar, Faten Chaieb, Hassen Drira, and Arezki Aberkane. Convivit–a deep neural network combining convolutions and factorized self-attention for human activity recognition. *arXiv preprint arXiv:2310.14416*, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2969–2978, 2022.

Christoph Feichtenhofer, Haoqi Fan, Bo Xiong, Ross B. Girshick, and Kaiming He. A large-scale study on unsupervised spatiotemporal representation learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3298–3308, 2021. URL `https://api.semanticscholar.org/CorpusID:233444206`.

Christoph Feichtenhofer, Yanghao Li, Kaiming He, et al. Masked autoencoders as spatiotemporal learners. *Advances in neural information processing systems*, 35:35946–35958, 2022.

Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3636–3645, 2017.

Nuno Cruz Garcia, Sarah Adel Bargal, Vitaly Ablavsky, Pietro Morerio, Vittorio Murino, and Stan Sclaroff. Distillation multiple choice learning for multimodal action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2755–2764, 2021.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pp. 5842–5850, 2017.

Tianyu Guo, Hong Liu, Zhan Chen, Mengyuan Liu, Tao Wang, and Runwei Ding. Contrastive learning from extremely augmented skeleton sequences for self-supervised action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 762–770, 2022.

Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *Advances in Neural Information Processing Systems*, 33:5679–5690, 2020.

Jian-Fang Hu, Wei-Shi Zheng, Jianhuang Lai, and Jianguo Zhang. Jointly learning heterogeneous features for RGB-D activity recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5344–5352, 2015.

Jian-Fang Hu, Wei-Shi Zheng, Lianyang Ma, Gang Wang, Jianhuang Lai, and Jianguo Zhang. Early action prediction by soft regression. *IEEE transactions on pattern analysis and machine intelligence*, 41(11): 2568–2583, 2018.

Bingkun Huang, Zhiyu Zhao, Guozhen Zhang, Yu Qiao, and Limin Wang. MGMAE: Motion guided masking for video masked autoencoding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13493–13504, 2023.

Hochul Hwang, Cheongjae Jang, Geonwoo Park, Junghyun Cho, and Ig-Jae Kim. Eldersim: A synthetic data generation platform for human action recognition in eldercare applications. *IEEE Access*, 2021.

Leyla Isik, Andrea Tacchetti, and Tomaso Poggio. A fast, invariant representation for human action in the visual system. *Journal of neurophysiology*, 119(2):631–640, 2018.

Jinhyeok Jang, Dohyung Kim, Cheonshu Park, Minsu Jang, Jaeyeon Lee, and Jaehong Kim. ETRI-activity3D: A large-scale rgb-d dataset for robots to recognize daily activities of the elderly. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10990–10997. IEEE, 2020.

Longlong Jing, Xiaodong Yang, Jingen Liu, and Yingli Tian. Self-supervised spatiotemporal feature learning via video rotation prediction. *arXiv preprint arXiv:1811.11387*, 2018.

Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

Boeun Kim, Hyung Jin Chang, Jungho Kim, and Jin Young Choi. Global-local motion transformer for unsupervised skeleton-based action learning. In *European conference on computer vision*, pp. 209–225. Springer, 2022.

Sangwon Kim, Dasom Ahn, and Byoung Chul Ko. Cross-modal learning with 3d deformable attention for action recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10265–10275, 2023.

Shi Lei, Zhang Yifan, Cheng Jian, and Lu Hanqing. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *CVPR*, 2019.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Unsupervised learning of view-invariant action representations. *Advances in neural information processing systems*, 31, 2018.

Lilang Lin, Sijie Song, Wenhan Yang, and Jiaying Liu. Ms2l: Multi-task self-supervised learning for skeleton based action recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, pp. 2490–2498, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379885. doi: 10.1145/3394171.3413548. URL https://doi.org/10.1145/3394171.3413548.

Chunhui Liu, Yueyu Hu, Yanghao Li, Sijie Song, and Jiaying Liu. PKU-MMD: A large scale benchmark for continuous multi-modal human action understanding. *arXiv preprint arXiv:1703.07475*, 2017.

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*, 42(10):2684–2701, 2020a.

Mengyuan Liu and Junsong Yuan. Recognizing human actions as the evolution of pose estimation maps. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 143–152, 2020b.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Yunyao Mao, Wengang Zhou, Zhenbo Lu, Jiajun Deng, and Houqiang Li. Cmd: Self-supervised 3d action representation learning with cross-modal mutual distillation. In *European Conference on Computer Vision (ECCV)*, 2022.

Vasu Parameswaran and Rama Chellappa. View invariance for human action recognition. *International Journal of Computer Vision*, 66(1):83–101, 2006.

AJ Piergiovanni and Michael S. Ryoo. Recognizing actions in videos from unseen viewpoints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4124–4132, June 2021.

Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6960–6970. IEEE, 2021.

Hossein Rahmani and Ajmal Mian. Learning a non-linear knowledge transfer model for cross-view action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2458–2466, 2015.

Cen Rao, Alper Yilmaz, and Mubarak Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50:203–226, 2002.

Arun V Reddy, Ketul Shah, William Paul, Rohita Mocharla, Judy Hoffman, Kapil D Katyal, Dinesh Manocha, Celso M de Melo, and Rama Chellappa. Synthetic-to-real domain adaptation for action recognition: A dataset and baseline performances. In *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023.

Dominick Reilly and Srijan Das. Just add?! pose induced video transformers for understanding activities of daily living. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18340–18350, 2024.

H. Saito, N. Inamoto, and S. Iwase. Sports scene analysis and visualization from multiple-view video. In *2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763)*, volume 2, pp. 1395–1398 Vol.2, 2004. doi: 10.1109/ICME.2004.1394492.

Madeline C Schiappa, Yogesh S Rawat, and Mubarak Shah. Self-supervised learning for videos: A survey. *ACM Computing Surveys*, 55(13s):1–37, 2023.

Younggyo Seo, Junsu Kim, Stephen James, Kimin Lee, Jinwoo Shin, and Pieter Abbeel. Multi-view masked world models for visual robotic manipulation. *arXiv preprint arXiv:2302.02408*, 2023.

Anshul Shah, Aniket Roy, Ketul Shah, Shlok Mishra, David Jacobs, Anoop Cherian, and Rama Chellappa. HaLP: Hallucinating latent positives for skeleton-based self-supervised learning of actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18846–18856, 2023a.

Ketul Shah, Anshul Shah, Chun Pong Lau, Celso M de Melo, and Rama Chellappa. Multi-view action recognition using contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3381–3391, 2023b.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+D: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1010–1019, 2016.

Xiangbo Shu, Jiawen Yang, Rui Yan, and Yan Song. Expansion-squeeze-excitation fusion network for elderly activity recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(8):5281–5292, 2022.

Nyle Siddiqui, Praveen Tirupattur, and Mubarak Shah. Dvanet: Disentangling view and action features for multi-view action recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 4873–4881, 2024.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pp. 568–576, 2014.

Xinyu Sun, Peihao Chen, Liangwei Chen, Changhao Li, Thomas H Li, Mingkui Tan, and Chuang Gan. Masked motion encoding for self-supervised video representation learning. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

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.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pp. 10347–10357. PMLR, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Shruti Vyas, Yogesh S Rawat, and Mubarak Shah. Multi-view action recognition using cross-view video prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pp. 427–444. Springer, 2020.

Dongang Wang, Wanli Ouyang, Wen Li, and Dong Xu. Dividing and aggregating network for multi-view action recognition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 451–467, 2018.

Jiang Wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2649–2656, 2014.

Jiangliu Wang, Jianbo Jiao, Linchao Bao, Shengfeng He, Yunhui Liu, and Wei Liu. Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4006–4015, 2019.

Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII 16*, pp. 504–521. Springer, 2020.

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*, pp. 14549–14560, 2023a.

Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Yu-Gang Jiang, Luowei Zhou, and Lu Yuan. BEVT: BERT pretraining of video transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14733–14743, 2022.

Rui Wang, Dongdong Chen, Zuxuan Wu, Yinpeng Chen, Xiyang Dai, Mengchen Liu, Lu Yuan, and Yu-Gang Jiang. Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6312–6322, 2023b.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14668–14678, 2022.

Lu Xia, Chia-Chih Chen, and Jake K Aggarwal. View invariant human action recognition using histograms of 3d joints. In *2012 IEEE computer society conference on computer vision and pattern recognition workshops*, pp. 20–27. IEEE, 2012.

Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10334–10343, 2019.

Siyuan Yang, Jun Liu, Shijian Lu, Meng Hwa Er, and Alex C. Kot. Skeleton cloud colorization for unsupervised 3d action representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13423–13433, October 2021.

Senthil Yogamani, Ciarán Hughes, Jonathan Horgan, Ganesh Sistu, Padraig Varley, Derek O'Dea, Michal Uricár, Stefan Milz, Martin Simon, Karl Amende, et al. Woodscape: A multi-task, multi-camera fisheye dataset for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9308–9318, 2019.

Pengfei Zhang, Jianru Xue, Cuiling Lan, Wenjun Zeng, Zhanning Gao, and Nanning Zheng. Adding attentiveness to the neurons in recurrent neural networks. In *proceedings of the European conference on computer vision (ECCV)*, pp. 135–151, 2018.

Jingjing Zheng, Zhuolin Jiang, P Jonathon Phillips, and Rama Chellappa. Cross-view action recognition via a transferable dictionary pair. In *bmvc*, volume 1, pp. 7, 2012.

Mohammadreza Zolfaghari, Yi Zhu, Peter Gehler, and Thomas Brox. Crossclr: Cross-modal contrastive learning for multi-modal video representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1450–1459, 2021.

A PyTorch code for motion weights

Given the input video frames, patch-size and the temperature parameter, we provide PyTorch code below for motion weights based on frame difference used in the paper.

Algorithm 1: PyTorch code for motion weights.

```
# frames    : input frames of shape [B,C,T,H,W]
# patch_size: (p_time, p_height, p_width)
# t         : temperature parameter

fdiff = frames[:,:,1:,:,:] - frames[:,:,:-1,:,:]
fdiff = torch.cat([fdiff[:,:,0:1,:,:], fdiff], dim=2)
fdiff = rearrange(fdiff,'b c (t p0) (h p1) (w p2) -> b (t h w) (p0 p1 p2 c)', p0=patch_size[0], p1=patch_size[1],
      p2=patch_size[2])
fdiff = torch.abs(fdiff)
fdiff = torch.linalg.vector_norm(fdiff, dim=2, keepdim=True) # B N 1
motion_weights = torch.nn.functional.softmax(fdiff/t, dim=1) # B N 1
```

B Architecture Details

The detailed asymmetric architecture of the encoder and decoders is shown in Table 11, Table 12 and Table 13. We have two decoders in our architecture: 1) self-view decoder and 2) cross-view decoder. The self-view decoder only uses self-attention to reconstruct same view whereas the cross-view decoder uses cross-attention in addition to self-attention for reconstructing target viewpoint while also using source viewpoint. These decoders are discarded during fine-tuning. We use 16 frame input and choose ViT-S/16 as our default encoder. We adopt the joint space-time attention (Arnab et al., 2021) for the encoder.

Stage	**Vision Transformer (Small)**	**Output Sizes**
data	stride 4×1×1	3×16×128×128
cube	2×16×16, 384 stride 2×16×16	384×**8**×64
mask	random mask *mask ratio* = ρ	384×**8**×[64×(1-ρ)]
encoder	$\begin{bmatrix} \text{MHA}(384) \\ \text{MLP}(1536) \end{bmatrix} \times 12$	384×**8**×[64×(1-ρ)]
projector	MLP(192) & *concat learnable tokens*	192×**8**×64

Table 11: **Encoder of MV2MAE.** The encoder processes 16-frame input clips from source and target views, and the encoded representations of the visible tokens are combined with the learnable mask tokens, before passing through the decoder.

C Additional Implementation Details

The pre-training and fine-tuning hyper-parameter settings for NTU-60, NTU-120 and ETRI datasets are given in Table 14 and Table 15.

Stage	Transformer	Output Sizes
self-view decoder	[MHA(192) MLP(768)] ×4	192×**8**×64
projector	MLP(1536)	1536×**8**×64
reshape	*from* 1536 *to* 3×2×16×16	3×16×128×128

Table 12: **Self-view decoder of MV2MAE.** It takes the source and target view tokens and reconstructs both the views independently.

Stage	Transformer	Output Sizes
cross-view decoder	[MHCA(192) MHA(192) MLP(768)] ×4	192×**8**×64
projector	MLP(1536)	1536×**8**×64
reshape	*from* 1536 *to* 3×2×16×16	3×16×128×128

Table 13: **Cross-view decoder of MV2MAE.** The cross-view decoder uses the visible tokens from the source view to reconstruct the missing patches in the target view.

config	NTU60	NTU120	ETRI
optimizer		AdamW	
base learning rate		1e-3	
weight decay		0.05	
optimizer momentum		β_1, β_2=0.9, 0.95	
batch size		1024	
learning rate schedule		cosine decay	
warmup epochs	320	160	160
total epochs	3200	1600	1600
augmentation		MultiScaleCrop	

Table 14: **Pre-training setting.**

config	NTU60	NTU120	ETRI
optimizer		AdamW	
base learning rate		1e-3	
weight decay		0.1	
optimizer momentum		β_1, β_2=0.9, 0.999	
batch size		1024	
learning rate schedule		cosine decay	
warmup epochs	5	10	10
training epochs	35	120	120
repeated augmentation		6	
flip augmentation		*yes*	
RandAug		(7, 0.5)	
label smoothing		0.1	
drop path		0.1	
layer-wise lr decay		0.9	

Table 15: **End-to-end fine-tuning setting.**

D Single-View vs Multi-View Inference

At test time, multiple viewpoints of an activity are available in the cross-subject setting. However, evaluation in prior work is carried out using single-view at a time, following the original benchmark (Shahroudy et al., 2016). Though in most practical scenarios, it would be natural to combine the predictions from available synchronized viewpoints for a given activity. We show this comparison of single-view and multi-view inference in Table 16. For multi-view inference, the predictions are combined using late fusion strategy.

Table 16: SV vs MV inference. We perform late fusion for multi-view inference.

	Cross-Subject (%)	
Method	NTU-60	NTU-120
Single-View Inference	90.0	85.3
Multi-View Inference	91.9	87.9