MOFO: MOtion FOcused Self-Supervision for Video Understanding

4th Workshop on Self-Supervised Learning: Theory and Practice
Anonymous Author(s)

Abstract

Self-supervised learning (SSL) techniques have recently produced outstanding results in learning visual representations from unlabeled videos. However, despite the importance of motion in supervised learning techniques for action recognition, SSL methods often do not explicitly consider motion information in videos. To address this issue, we propose MOFO (MOtion FOcused), a novel SSL method for focusing representation learning on the motion area of a video for action recognition. MOFO automatically detects motion areas in videos and uses these to guide the self-supervision task. We use a masked autoencoder that randomly masks out a high proportion of the input sequence and forces a specified percentage of the inside of the motion area to be masked and the remainder from outside. We further incorporate motion information into the finetuning step to emphasise motion in the downstream task. We demonstrate that our motion-focused innovations can significantly boost the performance of the currently leading SSL method (VidoeMAE) for action recognition. Our proposed approach significantly improves the performance of the current SSL method for action recognition, indicating the importance of explicitly encoding motion in SSL.

1 Introduction

Action recognition is an essential task in video understanding and has been extensively investigated in recent years Liu et al. [2022], Wei et al. [2022], Girdhar et al. [2022a]. In video action recognition, supervised deep learning techniques have made significant progress Tran et al. [2015], Feichtenhofer et al. [2019], Lin et al. [2019]; However, due to the lack of labels, which must be manually collected, learning to recognise actions from a small number of labelled videos is a difficult task as data collection will be expensive and challenging. It is especially inappropriate for long-tail open vocabulary object distributions across scenes, such as a kitchen. Furthermore, getting annotations for videos is much more difficult due to the large number of frames and the temporal boundaries of when actions begin and end.

Supervised methods Wang and Gupta [2018], Kwon et al. [2020], Patrick et al. [2021] have recognised the importance of motion to understand actions because often, key objects are moving in the scene. However, most SSL methods do not explicitly consider motion or use hand-crafted features Escorcia et al. [2022], limiting their effectiveness. In SSL literature, masked autoencoder models Tong et al. [2022] have been proposed to learn the underlying data distribution but without directly emphasising motion autonomously. Even though this model can perform spatiotemporal reasoning over content, the encoder backbone is ineffective in capturing motion representations (we show this later in Fig. 2). Incorporating motion information is not trivial. especially in egocentric video. The primary issue lies in the stability of the results, which can be significantly impacted by camera movement. When the camera moves rapidly, static objects or background pixels exhibit high movement velocities in optical flow. Several existing methods leveraged object detection to improve egocentric video recognition Wang et al. [2020b,b], Wu et al. [2019], Ma et al. [2016], among which Wu et al. [2019] also incorporate temporal contexts to help understand the ongoing action.

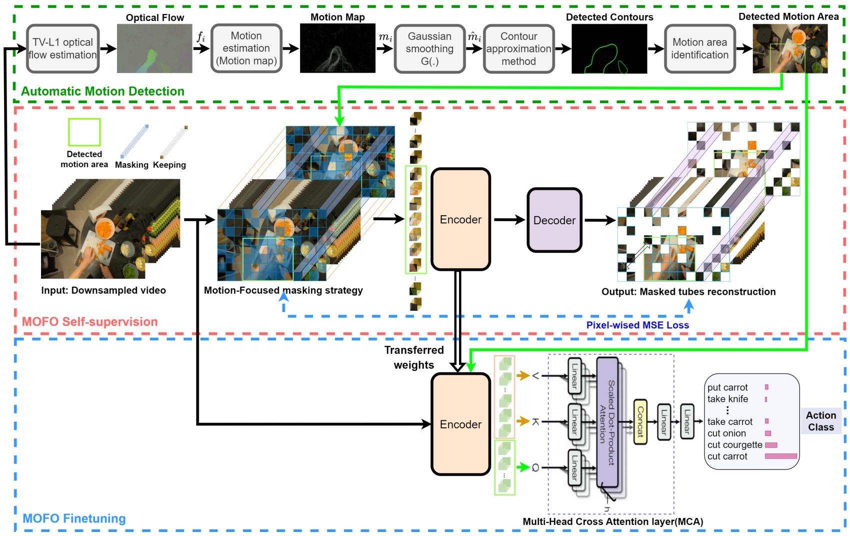


Figure 1: MOFO is a motion-focused self-supervised framework for action recognition.

These approaches may have limited uses in real-world systems since they demand time-consuming, labour-intensive item detection annotations and are computationally expensive. In contrast, our framework does not depend on costly object detectors.

Fig. 1 overviews our method, with three parts; first, our automatic motion area detection, With optical flow input to create a motion map to remove camera motion. Second, we propose our new strategy for the SSL pretext task, a reconstruction task focusing more on masking 3D patches on the motion area in the video called MOFO (Motion Focused). Thirdly, the downstream task adaptation step emphasises motion further by integrating motion information during the finetuning training. A key contribution of our work is to detect salient objects and motion in the video based on motion boundaries from optical flow. Using the motion boundaries instead of a direct optical flow output mitigates the challenge of camera motion and creates salient areas of movement or interest without a pretrained network. Given the identification of motion, we propose to provide a motion understanding of self-supervised masking Tong et al. [2022] of 3D patches in the video frames. A further contribution is that, during the finetuning stage, MOFO prioritises the motion areas in video data identified as a self-supervision pretext task. Since motion areas contain more information, such as moving objects, actions, and interactions, our proposed model gives them a higher priority by emphasising the masking strategy to be more in the motion area.

2 Motion-focused Self-supervised Video Understanding

2 .1 Automatic Motion Area Detection

To identify the motion areas without pretrained object detectors, we propose using classical computer vision features, Optical flow vectors; however, these vectors will be affected by camera motion, with static objects or background pixels exhibiting high movement velocities in optical flow when the camera moves rapidly. To mitigate the problem above, we calculate the motion boundaries Dalal et al. [2006] and use these to define a motion map Li et al. [2021]. Therefore, given a video with T frames and a $H \times W$ dimension, we first extract the optical flow vectors representing $\left\{f_i \in \mathbb{R}^{H \times W}\right\}_{i=1}^{T}$ pixel-level motion between two consecutive frames in a video using the TV-L1 algorithm Zach et al. [2007] that offers increased robustness against illumination changes, occlusions and noise. Then, given the horizontal and vertical displacements of each pixel between the ith frame and the $(i+1)$th frame represented by the flow maps $u_i, v_i \in \mathbb{R}^{H \times W}$, any kind of local differential or flow difference cancels out most of the effects of the camera rotation. The resulting motion map is defined as:

$$m_i = \sqrt{(\frac{\partial u_i}{\partial x})^2 + (\frac{\partial u_i}{\partial y})^2 + (\frac{\partial v_i}{\partial x})^2 + (\frac{\partial v_i}{\partial y})^2} \tag{1}$$

where every component denotes the corresponding x- and y-derivative differential flow frames contributing towards computing m_i, representing moving velocity in the i-th frame while ignoring the camera motion. As a result, $m_i \in \mathbb{R}^{H\times W}$ is less influenced by camera motion and considers the moving salients in the i-th frame. A low-pass Gaussian filter is used to smooth areas of the image with high-frequency components to further reduce the unwanted noise effect. The Gaussian Smoothing Operator computes an average of the surrounding pixels weighted according to the Gaussian distribution (G).

After noise reduction, the next step is to find the boundaries of the motion. To do so, we create contours Suzuki et al. [1985], which are short curves that connect points of the same hue or intensity. We select the two most significant contours in each frame to create a mask that indicates the motion area in a frame of a specific video. The main reason for choosing two contours is that in our datasets, an action is defined by hands and the corresponding object. We create a bounding box around the resulting area that precisely represents the motion in each video. In Fig. 7(a), we qualitatively compare our automatic box predictions and the provided supervised annotation for Epic-Kitchens-100 for several sample frames and provide further examples in the Appendix.

2 .2 Motion-focused Self-Supervised Learning

MOFO uses 3D tube volume embeddings for the self-supervised pretext stage to obtain 3D video patches from frames as inputs. It encodes these with a vanilla ViT Dosovitskiy et al. [2020] with joint space-time attention as a backbone. We segmented each video into N non-overlapping tubes $\mathbf{p}_i \in \mathbb{R}^{H_t\times W_t\times T_t}$. Then, we use a high-ratio tube masking approach to perform masked autoencoder (MAE) pretraining with an asymmetric transformer-based encoder-decoder architecture reconstruction task. Unlike other random masking methods, we explicitly integrate the motion information computed in subsection 2 .1 into our masking strategy, resulting in a motion-guided approach to encode motion for our MAE. Our novel tube masking strategy enforces a mask to be allied on a high portion of the tubes inside the motion area. In other words, a fixed percentage of the tubes (generally 75%) inside the motion area is always randomly masked to ensure the model is attending more to the motion area at reconstruction time. Therefore, we apply an extremely high ratio masking at random (90%) while always masking a fixed percentage of the tubes (75%) inside the motion area. The encoder produces a latent feature representation of the video using input frames with blacked-out regions. The decoder uses the latent feature representation from the encoder. It estimates the missing region using the mean squared error (MSE) loss, computed in pixel space between the masked patches and trained reconstructed outputs. Our design encourages the network to capture more useful spatiotemporal structures, making MOFO a more meaningful task and improving the performance of self-supervised pretraining. All models only use the unlabelled data in the training set of each dataset for pertaining.

2 .3 Motion-focused Finetuning

Recall that the self-supervised learning protocol is split between a pretraining and finetuning stage. We propose a new approach to focus on the motion area at both the pretext and the finetuning of the model. The model is trained end-to-end during finetuning, using the weights of the pretrained network as initialisation for the downstream supervised task dataset.

As the area inside the motion box has more semantic motion information, we wish to exploit this information for our task by leveraging the detected motion box. On the other hand, the video's setting and any nearby items could provide context for categorising the video clips for the action recognition task. For instance, in the case of washing dishes, the hands can be seen in the sink, but the dishes beside the sink may indicate that the person is washing them. Therefore, we propose to use multi-cross attention (MCA) Nagrani et al. [2021] in our encoder. MCA is an attention mechanism that mixes two different embedding sequences; the two are from the same modality. Unlike self-attention, where inputs are the same set, during cross-attention, they differ; MCA's main objective is to determine attention scores using data from various information sources. This module resides between the encoder and MLP classifier layers, takes the inner and outer motion box embeddings, and outputs the fused embedding (see details in Appendix B).

3 Experiments

We use two well-known and large datasets to evaluate our proposed approach: **Something-Something V2 (SSV2)** Goyal et al. [2017] and **Epic-Kitchens-100** Damen et al. [2022]. Using

Table 1: Human activity recognition on **Epic-Kitchens** and **Something-Something V2 (SSV2)** in terms of Top-1 and Top-5 accuracy.

Method	Backbone	Param	SSV2 Action Top-1	SSV2 Action Top-5	Epic-Kitchens Verb Top-1	Epic-Kitchens Noun Top-1	Epic-Kitchens Action Top-1
VIMPAC Tan et al. [2021]	ViT-L	307	68.1	-	-	-	-
BEVT Wang et al. [2022]	Swin-B	88	70.6	-	-	-	-
VideoMAE Tong et al. [2022]	ViT-B	87	70.8	92.4	71.6	66.0	53.2
ST-MAE Feichtenhofer et al. [2022]	ViT-L	304	72.1	-	-	-	-
OmniMAE Girdhar et al. [2022a]	ViT-B	87	69.5	-	-	-	39.3
Omnivore(Swin-B) Girdhar et al. [2022b]	ViT-B	-	71.4	93.5	69.5	61.7	49.9
MOFO (Proposed)	ViT-B	102	**75.5**	**95.3**	**74.2**	**68.1**	**54.5**

egocentric videos to predict first-person activity faces many challenges, including a limited field of view, occlusions, and unstable motions, and there is a relative scarcity of labelled data.

Results and analysis We finetune the learnt model for action classification based on our proposed MOFO finetuning approach to evaluate the learned model as a pretrained model and train on a new downstream task with the learned representation. The entire feature encoder and a linear layer are finetuned end-to-end with cross-entropy loss, with recognition accuracy reported in Table 3. We demonstrate significant performance improvement over the other self-supervised approaches, increasing 2.6%, 2.1%, and 1.3% accuracy over the best-performing methods on Epic-Kitchens verb, noun and action classification and 4.7% on Something Something V2 action classification, respectively. In terms of masking ratio, variants are presented in the Appendix, but we found that the 75% inside masking ratio worked the best. Our strategy outperforms approaches like OmniMAE Girdhar et al. [2022a], trained jointly on images and videos by 3.2% in Top-1 accuracy. On Something Something V2, our method outperforms VIMPAC Tan et al. [2021] and ST-MAE Feichtenhofer et al. [2022], which both use ViT-Large as a backbone, whereas our backbone is vanilla ViT-Base with over 3x fewer parameters. Compared to VideoMAE Tong et al. [2022], our approach achieves significantly better results while the number of backbone parameters remains the same.

Visualizing self-supervised representation To further understand the representations learnt by MOFO, we utilise GradCAM Selvaraju et al. [2017] to create a saliency map highlighting each pixel's importance to show how each pixel contributes to the discrimination of the video clip. Fig. 2 visualises the middle frame of a video clip, the motion map of the VideoMAE and our MOFO from the fifth attention layer of the ViT-Base backbone. It is interesting to note that for similar actions: *knead dough*, *cut carrot*, and *cut-in tomato*, MOFO is sensitive to the location that is the most significant motion location as detected by our automatic algorithm.

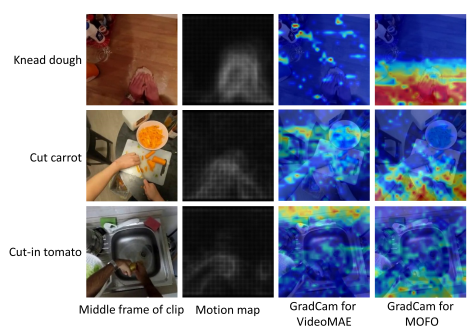


Figure 2: Visualisation of the learnt features

4 Conclusion

MOFO introduces a Motion-Focused technique, which explores the motion information for enhancing motion-aware self-supervised video action recognition. We propose an innovative strategy, an effective self-supervised pretext task, and a modification to masked autoencoding, which focuses masking on the motion area in the video (Motion Focused). Extensive experiments on two challenging datasets demonstrate that this context-based SSL technique improves performance in action recognition tasks, and the public code will guide many research directions.

References

Arif Akar, Ufuk Umut Senturk, and Nazli Ikizler-Cinbis. MAC: mask-augmentation for motion-aware video representation learning. In *BMVC*, 2022.

Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *ICCV*, 2021.

Federico Baldassarre and Hossein Azizpour. Towards self-supervised learning of global and object-centric representations. *arXiv preprint arXiv:2203.05997*, 2022.

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

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021.

Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In *NeurIPS*, 2020.

Yabo Chen, Yuchen Liu, Dongsheng Jiang, Xiaopeng Zhang, Wenrui Dai, Hongkai Xiong, and Qi Tian. Sdae: Self-distillated masked autoencoder. In *ECCV*, 2022.

Subhabrata Choudhury, Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Guess what moves: Unsupervised video and image segmentation by anticipating motion. *BMVC*, 2022.

Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, 2006.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *ECCV*, 2018.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. The epic-kitchens dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020a.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision. *arXiv preprint arXiv:2006.13256*, 2020b.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *IJCV*, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019.

Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, pages 1422–1430, 2015.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

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. *ICLR*, 2020.

Victor Escorcia, Ricardo Guerrero, Xiatian Zhu, and Brais Martinez. Sos! self-supervised learning over sets of handled objects in egocentric action recognition. In *ECCV*, 2022.

Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers. In *ICCV*, 2021.

Christoph Feichtenhofer, Axel Pinz, and Richard P Wildes. Spatiotemporal multiplier networks for video action recognition. In *CVPR*, 2017.

Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *ICCV*, 2019.

Christoph Feichtenhofer, haoqi fan, Yanghao Li, and Kaiming He. Masked autoencoders as spatiotemporal learners. In *NeurIPS*, 2022.

Basura Fernando, Hakan Bilen, Efstratios Gavves, and Stephen Gould. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017.

David J. Fleet and Allan D. Jepson. Stability of phase information. *IEEE TPAMI*, 1993.

Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, 2020.

Rohit Girdhar, Alaaeldin El-Nouby, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Omnimae: Single model masked pretraining on images and videos. *arXiv preprint arXiv:2206.08356*, 2022a.

Rohit Girdhar, Mannat Singh, Nikhila Ravi, Laurens van der Maaten, Armand Joulin, and Ishan Misra. Omnivore: A single model for many visual modalities. In *CVPR*, 2022b.

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 *ICCV*, 2017.

Sheng Guo, Zihua Xiong, Yujie Zhong, Limin Wang, Xiaobo Guo, Bing Han, and Weilin Huang. Cross-architecture self-supervised video representation learning. In *CVPR*, 2022.

Agrim Gupta, Stephen Tian, Yunzhi Zhang, Jiajun Wu, Roberto Martín-Martín, and Li Fei-Fei. Maskvit: Masked visual pre-training for video prediction. *arXiv preprint arXiv:2206.11894*, 2022.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

Muhammad Attique Khan, Kashif Javed, Sajid Ali Khan, Tanzila Saba, Usman Habib, Junaid Ali Khan, and Aaqif Afzaal Abbasi. Human action recognition using fusion of multiview and deep features: an application to video surveillance. *Multimedia tools and applications*, 2020.

Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM computing surveys (CSUR)*, 2022.

Heeseung Kwon, Manjin Kim, Suha Kwak, and Minsu Cho. Motionsqueeze: Neural motion feature learning for video understanding. In *ECCV*, 2020.

Conglong Li, Zhewei Yao, Xiaoxia Wu, Minjia Zhang, and Yuxiong He. Deepspeed data efficiency: Improving deep learning model quality and training efficiency via efficient data sampling and routing. *arXiv preprint arXiv:2212.03597*, 2022a.

Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. Semmae: Semantic-guided masking for learning masked autoencoders. *NeurIPS*, 2022b.

Haofeng Li, Guanqi Chen, Guanbin Li, and Yizhou Yu. Motion guided attention for video salient object detection. In *ICCV*, 2019.

Rui Li, Yiheng Zhang, Zhaofan Qiu, Ting Yao, Dong Liu, and Tao Mei. Motion-focused contrastive learning of video representations. In *ICCV*, 2021.

Yanghao Li, Chao-Yuan Wu, Haoqi Fan, Karttikeya Mangalam, Bo Xiong, Jitendra Malik, and Christoph Feichtenhofer. Mvitv2: Improved multiscale vision transformers for classification and detection. In *CVPR*, 2022c.

Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019.

Ze Liu, Jia Ning, Yue Cao, Yixuan Wei, Zheng Zhang, Stephen Lin, and Han Hu. Video swin transformer. In *CVPR*, June 2022.

Jonathon Luiten, Idil Esen Zulfikar, and Bastian Leibe. Unovost: Unsupervised offline video object segmentation and tracking. In *WACV*, 2020.

Minghuang Ma, Haoqi Fan, and Kris M Kitani. Going deeper into first-person activity recognition. In *CVPR*, 2016.

Joanna Materzynska, Tete Xiao, Roei Herzig, Huijuan Xu, Xiaolong Wang, and Trevor Darrell. Something-else: Compositional action recognition with spatial-temporal interaction networks. In *CVPR*, 2020.

Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion. *NeurIPS*, 2021.

Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016.

Adrián Núñez-Marcos, Gorka Azkune, and Ignacio Arganda-Carreras. Egocentric vision-based action recognition: A survey. *Neurocomputing*, 2022.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016.

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. *NeurIPS*, 2021.

Tomas Pfister, James Charles, and Andrew Zisserman. Flowing convnets for human pose estimation in videos. In *Proceedings of the IEEE international conference on computer vision*, 2015.

Sudeep Sarkar, P Jonathon Phillips, Zongyi Liu, Isidro Robledo Vega, Patrick Grother, and Kevin W Bowyer. The humanid gait challenge problem: Data sets, performance, and analysis. *IEEE transactions on pattern analysis and machine intelligence*, 2005.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017.

Ufuk Umut Senturk, Arif Akar, and Nazli Ikizler-Cinbis. Triplednet: Exploring depth estimation with self-supervised representation learning. 2022.

A H. Shabani, J S. Zelek, and D A. Clausi. Robust local video event detection for action recognition. In *NeurIPS, Machine Learning for Assistive Technology Workshop*, 2010.

Amir Hossein Shabani, David A Clausi, and John S Zelek. Improved spatio-temporal salient feature detection for action recognition. In *BMVC*, 2011.

Dandan Shan, Jiaqi Geng, Michelle Shu, and David F Fouhey. Understanding human hands in contact at internet scale. In *CVPR*, 2020.

Hedvig Sidenbladh, Michael J Black, and David J Fleet. Stochastic tracking of 3d human figures using 2d image motion. In *ECCV*, 2000.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *NeurIPS*, 2014.

S Sowmyayani and P Arockia Jansi Rani. Stharnet: Spatio-temporal human action recognition network in content based video retrieval. *Multimedia Tools and Applications*, 2022.

Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 1985.

Hao Tan, Jie Lei, Thomas Wolf, and Mohit Bansal. Vimpac: Video pre-training via masked token prediction and contrastive learning. *arXiv preprint arXiv:2106.11250*, 2021.

Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *NeurIPS*, 2022.

Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *TPAMI*, 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017.

Namrata Vaswani, Amit K Roy-Chowdhury, and Rama Chellappa. " shape activity": a continuous-state hmm for moving/deforming shapes with application to abnormal activity detection. *IEEE Transactions on Image Processing*, 2005.

Paul Viola, Michael J Jones, and Daniel Snow. Detecting pedestrians using patterns of motion and appearance. *IJCV*, 2005.

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 *CVPR*, 2019.

Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020a.

Lei Wang and Piotr Koniusz. Self-supervising action recognition by statistical moment and subspace descriptors. In *ACMMM*, 2021.

Limin Wang, Zhan Tong, Bin Ji, and Gangshan Wu. Tdn: Temporal difference networks for efficient action recognition. In *CVPR*, 2021.

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 *CVPR*, 2022.

Xiaohan Wang, Yu Wu, Linchao Zhu, and Yi Yang. Symbiotic attention with privileged information for egocentric action recognition. In *AAAI*, 2020b.

Xiaolong Wang and Abhinav Gupta. Videos as space-time region graphs. In *ECCV*, 2018.

Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *CVPR*, 2022.

Chao-Yuan Wu, Christoph Feichtenhofer, Haoqi Fan, Kaiming He, Philipp Krahenbuhl, and Ross Girshick. Long-term feature banks for detailed video understanding. In *CVPR*, 2019.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020.

Xuehan Xiong, Anurag Arnab, Arsha Nagrani, and Cordelia Schmid. M&m mix: A multimodal multiview transformer ensemble. *arXiv preprint arXiv:2206.09852*, 2022.

Yuwen Xiong, Mengye Ren, Wenyuan Zeng, and Raquel Urtasun. Self-supervised representation learning from flow equivariance. In *ICCV*, 2021.

Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019.

Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. In *CVPR*, 2022.

Charig Yang, Hala Lamdouar, Erika Lu, Andrew Zisserman, and Weidi Xie. Self-supervised video object segmentation by motion grouping. In *ICCV*, 2021.

Xitong Yang, Xiaodong Yang, Sifei Liu, Deqing Sun, Larry Davis, and Jan Kautz. Hierarchical contrastive motion learning for video action recognition. *arXiv preprint arXiv:2007.10321*, 2020.

Sukmin Yun, Hankook Lee, Jaehyung Kim, and Jinwoo Shin. Patch-level representation learning for self-supervised vision transformers. In *CVPR*, 2022.

Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Pattern Recognition: 29th DAGM Symposium, Heidelberg, Germany, September 12-14, 2007. Proceedings 29*, 2007.

Can Zhang, Yuexian Zou, Guang Chen, and Lei Gan. Pan: Persistent appearance network with an efficient motion cue for fast action recognition. In *ACMMM*, 2019.

Chuhan Zhang, Ankush Gupta, and Andrew Zisserman. Is an object-centric video representation beneficial for transfer? In *ACCV*, 2022.

Hao Zhang, Yanbin Hao, and Chong-Wah Ngo. Token shift transformer for video classification. In *ACMMM*, 2021.

Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

Appendix

We also conducted various ablation studies to examine the design choices made in our proposed strategy.

A Motion-focused Self-supervised Learning

Experimental setting. MOFO uses ViT-Base as a decoder/encoder backbone, trained for 800 epochs on Something-Something V2 and Epic-Kitchens for the SSL independently. We follow the training and experiential parameters from recent work Tong et al. [2022] to ensure a fair comparison and finetune for 100 epochs with early stopping. The model takes 16 frames from the video with 224×224 size and divides the input video into a 3D $16 \times 16 \times 8$ patch embeddings, resulting in $H = 224$, $W = 224$, $T = 16$, $H_t = 16$, $W_t = 16$, $T_t = 8$, and $N = 392$. While we have a fixed number of input patches for our model, we do not have a fixed number of inner N_{inner} and outer N_{outer} embeddings due to varying size of the motion area in each video clip. We report Top-1 accuracy on Epic-Kitchens and Top-1 and Top-5 accuracy on Something-Something V2 on downstream tasks and use Pytorch and DeepSpeed Li et al. [2022a] on 4xNVIDIA Quadro RTX-5000 GPU for our experiments.

Masking ratio. VideoMAE Tong et al. [2022] recommended tube masking with an extremely high ratio which helps reduce information leakage during masked modelling. They demonstrated the best efficiency and efficacy with a masking ratio of 90%. Therefore, we explore the effect of the inside masking ratio for verb classification on Epic-Kitchens in Fig. 3. It shows that the model pretrained with a masking ratio of 90% as the general masking ratio for a video and a high ratio for inside masking ratio (75%) achieves the highest efficiency level. Thus, we continue experimenting with the rest by fixing the inside mask ratio to 75%.

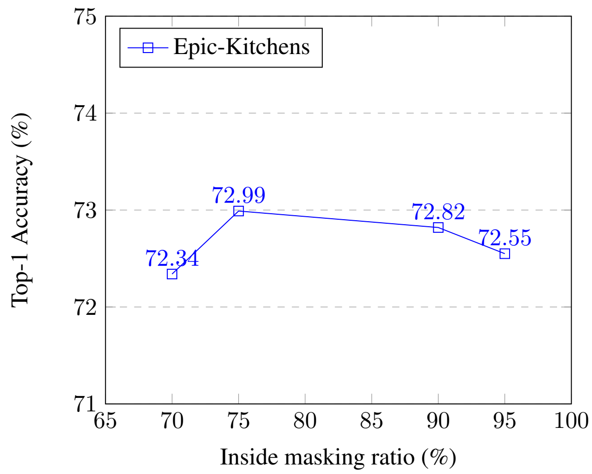


Figure 3: The effect of inside masking ratio on Epic-Kitchens-100 dataset for verb classification demonstrates that a high inside masking ratio (75%) delivers the best efficiency and effectiveness trade-off.

Reconstructed frames This section shows several reconstructed image frames from a video in Fig. 4 and Fig. 5. We use an asymmetric encoder-decoder architecture to accomplish video self-supervised pretraining tube masking with a high ratio for MAE pretraining. We can reconstruct the masked patches using random tube masking by finding the spatially and temporally corresponding unmasked patches in the adjacent frames. The loss function is the mean squared error (MSE) loss between normalised masked tokens and reconstructed tokens in pixel space. Videos are all randomly chosen from the validation sets of both datasets. Our proposed MOFO model ensures that a fixed number of masks exist within the motion area compared to the VideoMAE model. These examples suggest that, compared to VideoMAE, our MOFO model reconstructs the samples in the motion area significantly more accurately, demonstrating that the model has focused on the motion area. We can produce satisfying reconstruction results, mainly when motion occurs with our MOFO, by applying extremely high ratio masking at random (90%) while always masking a fixed percentage of the tubes (75%) inside the motion area.

B Motion-focused Finetuning

Setup Details Given a set of patches $\{\mathbf{p}_i\}_1^N$, the transformer yields two sets of embeddings: $\{\mathbf{e}^{\text{inner}}\}_{j=1}^{N_{\text{inner}}}$ for the inner motion boxes and $\{\mathbf{e}^{\text{outer}}\}_{k=1}^{N_{\text{outer}}}$ for the outer ones, as described by:

$$\{\mathbf{e}^{\text{inner}}\}_{j=1}^{N_{\text{inner}}}, \{\mathbf{e}^{\text{outer}}\}_{k=1}^{N_{\text{outer}}} = \text{ViT}\big(\{\mathbf{p}_i\}_1^N\big) \tag{2}$$

These embeddings are then processed by a cross-attention mechanism, where Q, K, and V represent query, key, and value, respectively. The CrossAttention function is formalised as follows:

$$\text{CrossAttention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

where $Q = \mathbf{e}^{\text{inner}}, K = V = \mathbf{e}^{\text{outer}}$. In the context of multi-head attention, each attention head i is computed by applying the CrossAttention function to the query, key, and value matrices, each weighted by a different learned weight matrix $W_i^Q \in \mathbb{R}^{d_{model} \times d_q}, W_i^k \in \mathbb{R}^{d_{model} \times d_k}, W_i^V \in \mathbb{R}^{d_{model} \times d_v}$ respectively:

$$\text{head}_i = \text{CrossAttention}(QW_i^Q, KW_i^K, VW_i^V) \tag{4}$$

Finally, the fused embedding $\mathbf{e}^{\text{fused}}$ is computed by concatenating the results from all attention heads and then applying another learned weight matrix $W^O \in \mathbb{R}^{hd_v \times d_{model}}$. This multi-head cross-

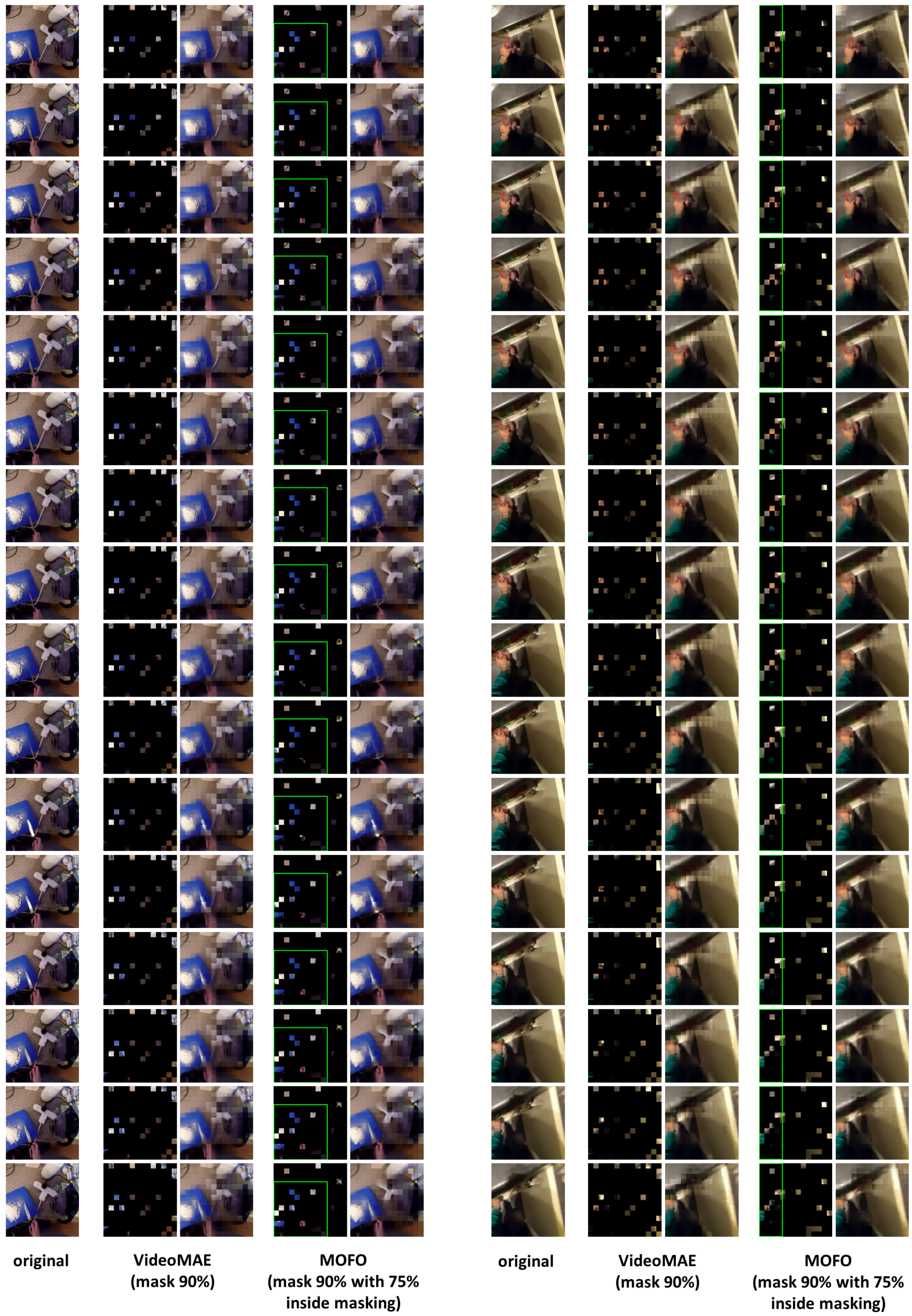


Figure 4: Qualitative Comparison on reconstructions using VideoMAE and MOFO on **Epic-Kitchens** dataset. MOFO Reconstructions of videos are predicted by MOFO pre-trained with a masking ratio of 90% and an inside masking ratio of 75% .

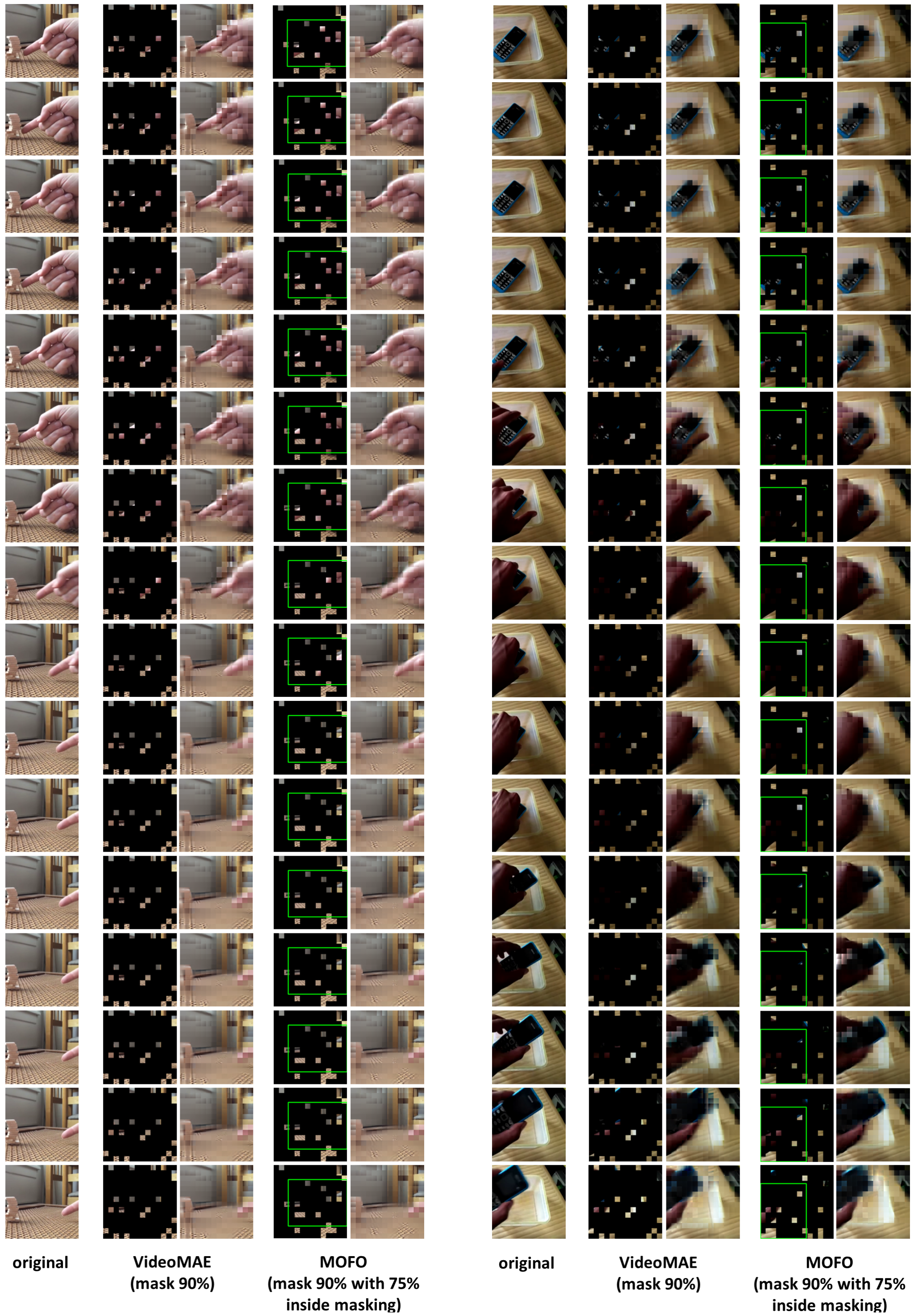


Figure 5: Qualitative Comparison on reconstructions using VideoMAE and MOFO on **Something-Something V2** dataset. MOFO Reconstructions of videos are predicted by MOFO pre-trained with a masking ratio of 90% and an inside masking ratio of 75%.

attention (MCA) operation can be represented as:

$$\mathbf{e}^{\text{fused}} = \text{MCA}(Q, K, V) = \text{Concat}\left(\text{head}_1, \cdots, \text{head}_h\right) W^O \tag{5}$$

We employ $h = 3$ parallel attention layers, or heads, in this work. We also use $d_q = d_k = d_v = d_{model}$ for each. The model is ultimately finetuned with a cross-entropy loss $\mathcal{L}$:

$$\begin{aligned} \mathcal{L} &= -\sum_n \mathbf{y}_n \log \hat{\mathbf{y}}_n \\ \hat{\mathbf{y}} &= \text{FC}(\mathbf{e}^{\text{fused}}) \end{aligned} \tag{6}$$

where, $\mathbf{y}_n$ is the true label for nth video clip, $\hat{\mathbf{y}}_n$ is its predicted label, and FC is the fully connected layers typically used for classification.

MCA hyper-parameters ablation. We list the MCA hyperparameters used in our MOFO finetuning experiments here. We experiment with various head and depth settings when Epic-Kitchens is the target dataset shown in Table 2. We experiment with these parameters for the verb task on Epic-Kitchens to find the best choice for the cross-attention layer we suggested for MOFO finetuning. The final head and depth are 3 and 1, respectively.

Table 2: Ablation experiment for number of head and depth in MOFO finetuning

Finetuning method	Backbone training	CA heads	CA depths	**Epic-Kitchens** Verb Top-1
VideoMAE	VideoMAE	-	-	71.6
MOFO	VideoMAE	1	1	73.5
MOFO	VideoMAE	1	2	73.8
MOFO	VideoMAE	1	3	73.6
MOFO	VideoMAE	2	1	73.7
MOFO	VideoMAE	2	2	73.3
MOFO	VideoMAE	**3**	**1**	**74.0**
MOFO	VideoMAE	3	2	73.5
MOFO	VideoMAE	4	1	73.8
MOFO	VideoMAE	4	2	73.3

Visualisation of GradCAM using MOFO self-supervision We visualise the GradCAM and motion map in Fig. 6 for the samples in which VideoMAE can't identify the class, but our MOFO can. The attention maps show how effective our approach is in capturing the motion area. Visualisation of important areas. The heatmap indicates how much the pretrained model attends to the region.

C Ablation Study

We finetune the learnt model for action classification to evaluate the learned model as a pretrained model and train on a new downstream task with the learned representation. We perform such an evaluation on our self-supervised model to gain some insights into the generality of the learned features. For finetuning, we follow the same protocol in Tong et al. [2022] to provide a fair comparison and call it regular finetuning. The entire feature encoder and a linear layer are finetuned end-to-end with cross-entropy loss, The recognition accuracy for our MOFO SSL using regular finetuning is reported in Table 3 shown as MOFO*. We demonstrate significant performance improvement over the other self-supervised approaches, comparable to the best-supervised approach. All variants of our model are presented in section A outperformed the existing result using ViT-MAE, but we found that the 75% inside masking ratio worked the best. Compared to VideoMAE Tong et al. [2022], our approach achieves significantly better results while the number of backbone parameters remains the same. While MOFO** indicates our result with pretraining on non-motion SSL and MOFO finetuning, which further increases accuracy, MOFO† denotes the MOFO SSL and MOFO finetuning, which we mention in Table 3 as MOFO(Proposed), and this provides the greatest performance over the best-performing methods on Epic-Kitchens verb, noun and action classification and on Something Something V2 action classification.

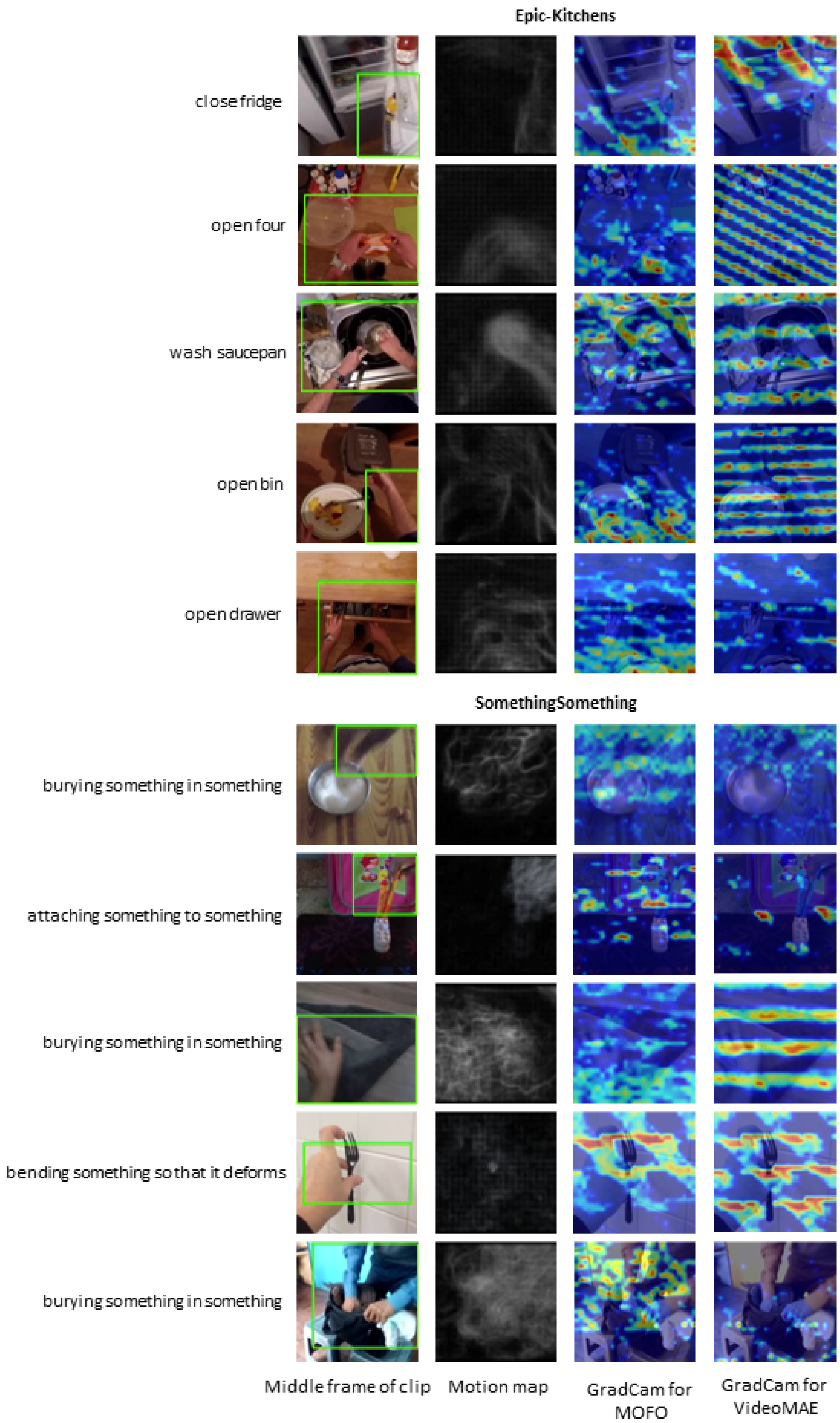


Figure 6: We visualise the attention maps generated by GradCAM based on VideoMAE and MOFO for Epic-Kitchens and the Something-Something V2 dataset. The attention maps show that our proposed approach can better capture the motion area.

Table 3: Human activity recognition on **Epic-Kitchens** and **Something-Something V2 (SSV2)** in terms of Top-1 and Top-5 accuracy. blue: This is the result computed by us using the public code MOFO* is pretrained by our MOFO SSL and uses non-MOFO finetuning. MOFO** This is our result with pretraining on non-MOFO SSL and has MOFO finetuning. MOFO† denotes the MOFO SSL and MOFO finetuning.

Method	Backbone	Param	SSV2 Action Top-1	SSV2 Action Top-5	Epic-Kitchens Verb Top-1	Epic-Kitchens Noun Top-1	Epic-Kitchens Action Top-1
Supervised							
TDN$_{EN}$ Wang et al. [2021]	ResNet101Œ2	88	69.6	92.2	-	-	-
SlowFast Feichtenhofer et al. [2019]	ResNet101	53	63.1	87.6	65.6	50.0	38.5
TSM Lin et al. [2019]	ResNet-50	-	63.4	88.5	67.9	49.0	38.3
MViTv1 Fan et al. [2021]	MViTv1-B	37	67.7	90.9		-	-
TimeSformer Bertasius et al. [2021]	ViT-B	121	59.9	-	-	-	-
TimeSformer Bertasius et al. [2021]	ViT-L	430	62.4	-	-	-	-
ViViT FE Arnab et al. [2021]	ViT-L	-	65.9	89.9	66.4	56.8	44.0
Mformer Patrick et al. [2021]	ViT-B	109	66.5	90.1	66.7	56.5	43.1
Mformer Patrick et al. [2021]	ViT-L	382	68.1	91.2	67.1	57.6	44.1
Video SWin Liu et al. [2022]	Swin-B	88	69.6	92.7	67.8	57.0	46.1
Self-supervised							
VIMPAC Tan et al. [2021]	ViT-L	307	68.1	-	-	-	-
BEVT Wang et al. [2022]	Swin-B	88	70.6	-	-	-	-
VideoMAE Tong et al. [2022]	ViT-B	87	70.8	92.4	71.6	66.0	53.2
ST-MAE Feichtenhofer et al. [2022]	ViT-L	304	72.1	-	-	-	-
OmniMAE Girdhar et al. [2022a]	ViT-B	87	69.5	-	-	-	39.3
Omnivore(Swin-B) Girdhar et al. [2022b]	ViT-B	-	71.4	93.5	69.5	61.7	49.9
Ours(MOFO*)	ViT-B	87	**72.7**	**94.2**	**73.0**	**67.1**	**54.1**
Ours(MOFO)**	ViT-B	102	**74.7**	**95.0**	**74.0**	**68.0**	**54.5**
Ours(MOFO†)	ViT-B	102	**75.5**	**95.3**	**74.2**	**68.1**	**54.5**

Table 4: Human activity recognition on **Epic-Kitchens** and **Something-Something V2** in terms of Top-1 accuracy. blue: This is the result computed by us using the public code MOFO* is pretrained by our MOFO SSL and uses non-MOFO (regular) finetuning.

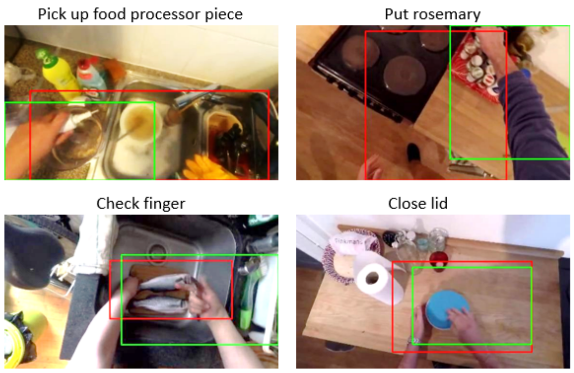
Method	Backbone	Pretrain Dataset	Something-Something V2 Action Top-1	Epic-Kitchens Verb Top-1	Epic-Kitchens Noun Top-1	Epic-Kitchens Action Top-1
VideoMAE Tong et al. [2022]	ViT-B	$Something-SomethingV2$	70.8	70.2	62.9	50.7
VideoMAE Tong et al. [2022]	ViT-B	$Epic-Kitchens$	67.3	71.6	66.0	53.2
Ours(MOFO*)	ViT-B	$Something-SomethingV2$	72.7	70.0	62.7	50.6
Ours(MOFO*)	ViT-B	$Epic-Kitchens$	67.4	73.0	67.1	54.1

D Domain Generalization

Domain generalisation aims to build a predictor that can perform well in an unseen test domain, known as out-of-distribution generalisation. The main objective of this experiment is to learning video representations that transfer well to a novel previously unseen dataset. We take the MOFO and non-MOFO pretrained models that have already learned features from one dataset and finetune them to adapt them to a new dataset. Results in Table **??** show that our proposed MOFO model and non-MOFO pretrained model got on-par results; our MOFO pretrained model's accuracy on SSV2 is marginally higher when pretraining is done on Epic-Kitchens, and marginally worse on Epic-Kitchens when pretraining is done on SSV2. These results have inspired me to design a self-supervision task to enhance generalisation.

E Automatic Motion Area Detection

Automatic vs. supervised motion area detection. We compare the results using our automatically detected motion areas and the ground truth bounding box annotation provided by Damen et al.



(a)

Method	Annotation	**Epic-Kitchens** Verb Top-1
MOFO supervision	Supervised	73.26
	Automatic(ours)	72.99

(b)

Figure 7: (a) Comparison between the unsupervised and supervised motion area detection, green rectangles indicate the unsupervised while red ones show supervised detected motion area. (b) Effect of supervised vs. automatic motion area utilisation in MOFO.

[2022] on the Epic-Kitchens dataset in Table 7(b). Our automatic motion detection results are close compared to supervised annotations, as seen in Table 7(b), despite the challenging camera motion from the egocentric videos.

We compute the Intersection over the Union (IoU) metric to compare our automatic detector with the supervised annotated bounding boxes on both datasets Damen et al. [2022], Materzynska et al. [2020]. For the Epic-Kitchens dataset, the IoU is 40%, and for Something-Something V2, the IoU is 31%. Although these numbers are lower, our automatic motion detection only detects motion and ignores unnecessary static objects near the motion. As you can see in Fig. 7(a), our automatic motion box still focuses on the area and object of interest, which is the key requirement.

In Fig. 8, we present additional qualitative examples of our automatic motion area detection compared with the provided supervised annotation for Epic-Kitchens and Something-Something V2 datasets. These samples show that our proposed automatic motion area detection minimises the impact of the static object in the motion box while highlighting the motion areas. Our automatic motion box concentrates on the area and item of interest, which is necessary for our proposed approach, even for self-supervision or finetuning.

F Related Work

Self-supervised learning (SSL) is a developing machine learning technique that has the potential to address the issues brought about by over-dependenceăon labelled data. High-quality labelled data have been essential for many years to develop intelligent systems using machine learning techniques. Consequently, high-quality annotated data costs are a significant bottleneck in the training process. Grow the research and development of generic AI systems at an inexpensive cost. Self-learning mechanisms with unstructured data are one of the top focuses of AI researchers. Collecting and labelling a wide range of diverse data is almost impossible. Researchers are developing self-supervised learning (SSL) methods that can pick up on fine details in data to address this issue. The introduction to self-supervised learning in video understanding is followed by a review of the literature on video action recognition, the downstream task we have recently focused on.

F .1 Self-supervised Video Representation learning

The effectiveness of deep learning-based computer vision relies on the availability of a considerable amount of annotated data, which is time-consuming and expensive to obtain. Supervised learning is trained over a given task with a large, manually labelled dataset. In addition to the costly manual labelling, generalisation mistakes and erroneous correlations are other problems with supervised learning.

Large labelled datasets are difficult to create in particular situations, making it challenging to construct computer vision algorithms. Most computer vision applications in the real world use visual categories not included in a common benchmark dataset. In specific applications, visual categories or their appearance are dynamic and vary over time. Therefore, self-supervised learning could be

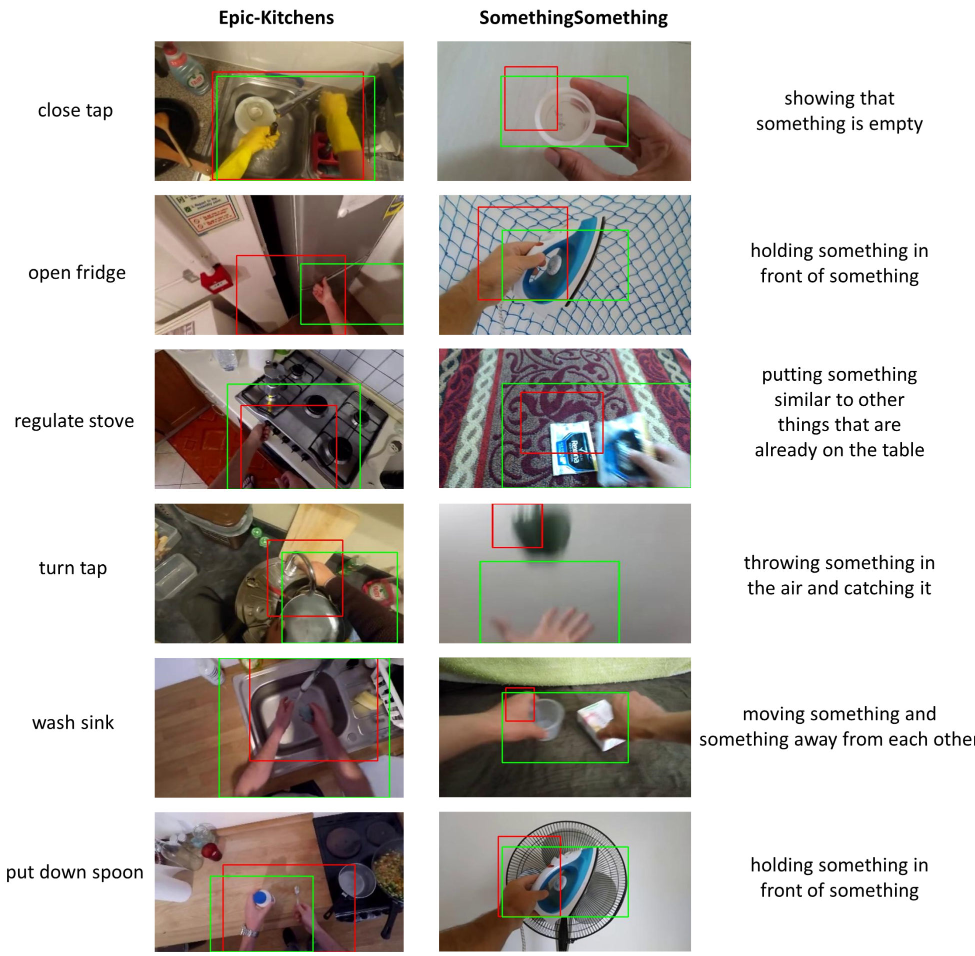


Figure 8: Comparison between the unsupervised and supervised motion area detection, green rectangles indicate the unsupervised while red ones show supervised detected motion area.

created that uses a limited number of labelled examples to learn to recognise new concepts effectively. A substantial research effort focuses on learning from unlabeled data, which is much easier to acquire in real-world applications. The ultimate goal is to make it possible for machines to comprehend new concepts quickly after only viewing a few labelled instances, similar to how quickly humans can learn.

SSL has gained considerable popularity since its introduction in natural language processing Devlin et al. [2019] and computer vision Doersch et al. [2015], Chen et al. [2020], Xie et al. [2020] owing to its ability to learn effective data representations without requiring manual labels. Acquiring detailed manual labels is arguably more difficult (and often expensive) in many image and video-related tasks, which makes SSL an increasingly popular paradigm in video analysis.

The goal of video self-supervised learning for computer vision is to learn meaningful video representations without explicit supervision, and the model trains itself to learn one part of the input from another part of the input. Self-supervised learning algorithms can learn representations by solving pretext tasks that can be formulated using only unlabeled data. These auxiliary tasks can guide the model to learn intermediate representations of data. By solving these tasks, the model learns to extract relevant features from the input data and understand the underlying structural meaning beneficial for practical downstream tasks. Based on the surrogate task employed, the training objective for self-supervised learning is defined, and model parameters are updated through gradient descent to minimise prediction error. Therefore, models are trained to solve these pretext tasks. As a result, they learn to capture meaningful and useful representations that can be used for various downstream video understanding tasks, such as video action recognition F .2.

Video-based self-supervised learning techniques start from image tasks. Several specifically designed tasks, including image inpainting Pathak et al. [2016], solving jigsaw puzzles Noroozi and Favaro [2016], and image colour channel prediction Zhang et al. [2016] are proposed to learn image features. SSL has recently yielded successful results in learning visual representations from unlabeled videos with various pretext tasks Yun et al. [2022], Caron et al. [2021], Gupta et al. [2022]. These methods use a backbone that has been pretrained with images or videos in a self-supervised manner to perform tasks on videos, including contrastive learning Yun et al. [2022], Guo et al. [2022], Yang et al. [2020], self-distillation Caron et al. [2021], or Masked Modeling which selects a random section of the input sequence to mask out, and then predicts the features of those sections Wei et al. [2022], Gupta et al. [2022], Tong et al. [2022], Girdhar et al. [2022a]. Many existing works Fernando et al. [2017], Xu et al. [2019], Wang et al. [2020a] have been proposed to focus on temporal information, such as making models sensitive to the temporal differences of input data.

As mentioned before, earlier works build on a concept of self-supervision by taking RGB frames as input to learning to predict action concepts Wang and Koniusz [2021], using Convolutional Neural Networks (CNNs) models to use frame-wise features and average pooling Karpathy et al. [2014] discarding the temporal order. Thus, frame-wise CNN scores were fed to LSTMs Donahue et al. [2015] while in two-stream networks Simonyan and Zisserman [2014], representations are computed for each RGB frame and every ten stacked optical flow frames. Spatio-temporal 3D CNN filters Tran et al. [2015], Varol et al. [2017], Feichtenhofer et al. [2017], Carreira and Zisserman [2017] model spatio-temporal patterns.Persistence of Appearance, a motion cue proposed by PAN Zhang et al. [2019], allows the network to extract the motion information from adjacent RGB frames directly. Vision Transformers (ViTs) Dosovitskiy et al. [2020], Khan et al. [2022] have emerged as an effective alternative to traditional CNNs. The architecture of Vision Transformer is inspired by the prominent Transformer encoder Devlin et al. [2018], Vaswani et al. [2017] used in natural language processing (NLP) tasks, which process data in the form of a sequence of vectors or tokens. Like the word tokens in NLP Transformer, ViT generally divides the image into a grid of non-overlapping patches before sending them to a linear projection layer to adjust the token dimensionality. Feed-forward and multi-headed self-attention layers are then used to process these tokens. ViTs have a wide range of applications in numerous tasks due to their capacity to capture global structure through self-attention, such as classification Zhang et al. [2021], Xiong et al. [2022], Li et al. [2022b], object detection Chen et al. [2022], Li et al. [2022c], segmentation Choudhury et al. [2022], Caron et al. [2021], Baldassarre and Azizpour [2022] and retrieval Gabeur et al. [2020].

Inspired by ViT Dosovitskiy et al. [2020], ViViT Arnab et al. [2021] and Timesformer Bertasius et al. [2021] were the first two works that successfully implemented a pure transformer architecture for video classification, improving upon the state of the art previously set by 3D CNNs. In these models, the video clip of RGB frames is embedded into 3D patches to produce downsampled feature maps. Then, these encoded 3D patches are encoded by a Video Transformer Patrick et al. [2021], Zhang et al. [2022]. In the following work, Arnab et al. [2021] defines the tubelet embedding tokenisation method and inspired some other works to represent a video input by extracting non-overlapping, spatiotemporal tubes to propose their method Yan et al. [2022].

In another line of research, Masked Autoencoders (MAEs) have recently been demonstrated to be powerful yet conceptually simple and efficient and have proven an effective pretraining paradigm for Transformer models of text Devlin et al. [2018], images He et al. [2022], and, more recently, videos Tong et al. [2022]. The learnt self-supervised model from the pretext task can be applied to any downstream computer vision tasks, including classification, segmentation, detection, etc.

Nowadays, encoder-decoder Transformer-based architectures are commonly used in self-supervised learning for video representation learning. These architectures take advantage of the Transformer models' strengths, initially created for natural language processing challenges, and adapt them to process and comprehend video data. In the context of video representation learning, the encoder-decoder Transformer architecture typically consists of the following components:

1. **Encoder** The encoder processes the input video data and generates a condensed representation of the video. Each video frame or 3D tublets is typically treated as a sequence of features to be input into the Transformer encoder. Multiple layers of self-attentional and feed-forward neural networks can be used in the encoder to capture the video's temporal dependencies, spatial relationships, and long-range dependencies.

2. **Decoder:** Based on the self-supervised task, the decoder generates a prediction using the encoder's learnt representation. The decoder must solve the surrogate task used for self-

supervised learning. For instance, if the self-supervised objective is to anticipate the temporal order of shuffled frames, the decoder may correctly predict that order.

In transformer-based architecture, the self-attention mechanism powers both the encoder and decoder. Self-attention architectures typically are made up of a series of transformer blocks. Each transformer block consists of two sublayers: a feed-forward layer and a multi-head self-attention layer. An input is divided into patches, and attention evaluates each 3D input patch's usefulness before drawing on it to produce the output. The Transformer's self-attention mechanism lets the model focus on different parts of the video frames while considering their dependencies. Therefore, considering their relative importance, it draws from each input component to produce the output. The query(Q), key(K), and value(V) vectors are the three sets of calculated vectors in the transformer architecture. These are determined by multiplying the input by a linear transformation.

F .2 Video Action Recognition

Although it is simple for humans to recognise and categorise actions in video, automating this process is challenging. Human action recognition in video is of interest for applications such as automated surveillance Khan et al. [2020] detecting anomalies in a cameras field of view that has attracted attention from vision researchers Vaswani et al. [2005], elderly behaviour monitoring Sarkar et al. [2005], human-computer interaction, content-based video retrieval Sowmyayani and Rani [2022], and video summarization Shabani et al. [2011]. Activity analysis must be able to identify atomic movements like "walking," "bending," and "falling" on their own while monitoring the daily activities of elderly people, for instance Shabani et al. [2010]. Therefore, action recognition is a challenging problem with many potential applications.

Action Recognition Datasets Human action recognition aims to understand human activities occurring in a video as humans can understand. While some simple actions, like standing, can be recognised from a single frame (image), most human actions are much more complex and occur over a more extended period. Therefore, they must be observed through consecutive frames (video). To assist organisations in understanding real-time action and dynamic, organic movement, AI/ML models use human actionădatasets.

Something-Something V2 Goyal et al. [2017] This publically available dataset is an extensive collection of human-object interaction of densely labelled 174 video sequences. The dataset was created by many crowd workers performing pre-trained daily humanobject interaction physical activities; 220,847 videos and JPG images have variable spatial resolutions and lengths.

Egocentric vision, sometimes known as first-person vision, is a sub-field of computer vision that deals with analysing images and videos captured by a wearable camera, often worn on the head or the chest and thus naturally approximates the wearer's visual field. The idea of using egocentric videos has recently been utilised thanks to novel, lightweight and affordable devices such as GoPro and similars Núñez-Marcos et al. [2022]. As a fundamental problem in egocentric vision, one of the tasks of egocentric action recognition aims to recognise the actions of the camera wearers from egocentric videos. This community did not have an extensive dataset to be used for pertaining or to have a standard dataset for benchmarking until the appearance of the Epic-Kitchens Damen et al. [2018, 2020a,b], the largest and most complete egocentric dataset contains 97 verb classes, 300 noun classes and 3806 action classes. Understanding egocentric videos requires detecting the actor's movement and the object with which the actor interacts.

Several existing methods leveraged object detection to improve egocentric video recognition Wang et al. [2020b,b], Wu et al. [2019], Ma et al. [2016], among which Wu et al. [2019] also incorporate temporal contexts to help understand the ongoing action. These approaches may have limited uses in real-world systems since they demand time-consuming, labour-intensive item detection annotations and are computationally expensive. In contrast, our framework does not depend on costly object detectors. Recently, Shanetal.Shan et al. [2020] developed a hand-object detector to locate the active object. When the detector is well-trained, it can be deployed on the target dataset; however, running it on high-resolution frames still costs far more than using our method.

Motion in Action Recognition: Motion cuesAkar et al. [2022], Wang et al. [2019], Li et al. [2021] have been recognised as necessary for video understanding in the past few years. Most works use optical flow, a motion representation component in many video recognition techniques, to obtain the statistical motion labels required for their work Yang et al. [2021], separating the background from the main objects in optical flow frames. Optical flow is the pattern of visible motion of objects and edges and helps calculate the motion vector of every pixel in a video frame. Optical flow is widely used in many video processing applications as a motion representation feature that can give important information about the spatial arrangement of the objects viewed and the rate of change of this arrangement. Optical flow-based techniques are sensitive to camera motion since they capture absolute movement. Optical flow computation is one of the fundamental tasks in computer vision. In practice, the flow has been helpful for a wide range of problems, for example, pose estimation Pfister et al. [2015], representation learning Senturk et al. [2022], segmentation Luiten et al. [2020], and even utilised as a tracking substitute for visual signals (RGB images) Sidenbladh et al. [2000]. Since optical flow can capture continuous or smoothly varying motion, such as motion caused by a change in camera view, it is not a good idea to use it to detect a change in salient objects. To build pixel-level representations from raw high-resolution videos with complex scenes, Xiong et al. [2021] proposes a self-supervised representation learning framework based on a flow equivariance objective. This representation is beneficial for object detection. In another work Li et al. [2019], a multi-task motion-guided video salient object detection network is proposed consisting of two sub-networks. One sub-network is used to detect salient objects in still images, and the other is used to detect motion saliency in optical flow images. Most motion descriptors use absolute motions and thus only work well when the camera and background are relatively static, such as Fleet & Jepson's phase-based features Fleet and Jepson [1993] and Viola et al.'s generalised wavelet features Viola et al. [2005]. Therefore, the critical problem is identifying characteristics that accurately capture the motion of hands or objects while impervious to the camera and backdrop motion.

Relying only on optical flow to capture the motion is not a robust solution as it is heavily affected by camera motion. To mitigate this problem, Wang et al. [2019] presented a self-supervised spatiotemporal video representation by predicting a set of statistical labels derived from motion and appearance statistics using extracting optical flow across each frame and two motion boundaries Dalal et al. [2006] which are obtained by computing gradients separately on the horizontal and vertical components of the optical flow.

In another line of work, masked autoencoder models have been proposed to learn underlying data distribution in a self-supervised manner without explicitly focusing on motion Tong et al. [2022]. Even though this model can perform spatiotemporal reasoning over content, the encoder backbone could be more effective in capturing motion representations. The critical contribution of our work is explicitly imposing motion information in both SSL phases in the self-supervised pretext training without human annotations and then in the finetuning stage, besides introducing an automatic motion detection to detect salient objects and motion in the video without the overhead and limitation of a pretrained and annotated object detector.