Supplementary: On the Utility of 3D Hand Poses for Action Recognition

Md Salman Shamil[1], Dibyadip Chatterjee[1], Fadime Sener[2], Shugao Ma[2], and Angela Yao[1]

[1] National University of Singapore
[2] Meta Reality Labs Research
https://s-shamil.github.io/HandFormer/

A Statistical Analysis: Full-body vs. Hand Skeletons

This section serves as an extension of Sec. 3 to statistically analyze the differences between hand poses and full-body poses. Action recognition datasets [7,11], which primarily focus on full-body actions, often include actions involving partial body movements. These actions exhibit limited global motion when viewed

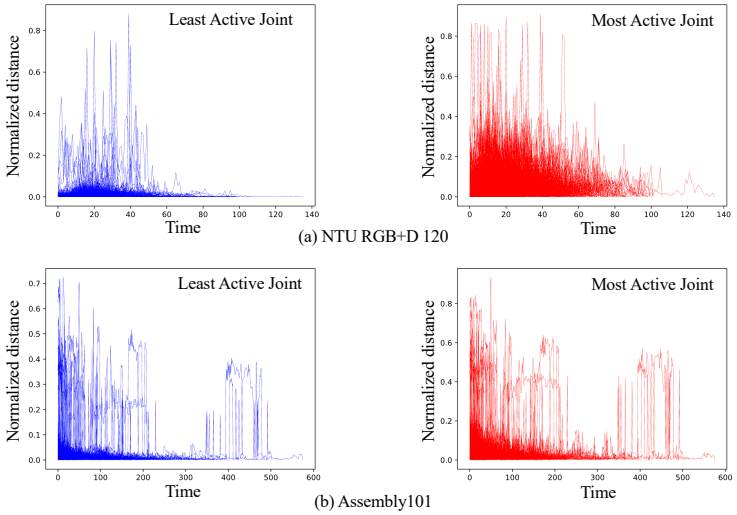


Fig. 1: With a random pool of 1000 sequences, we observe that the least active joints can be viewed as static reference points, showing minimal movement in NTU RGB+D 120. In contrast, Assembly101 exhibits subtler distinctions between the most active and the least active joints. The Pearson correlation coefficient (r) between the distance values for these two joints yields a high value (0.93) for Assembly101, while $r = 0.33$ for NTU RGB+D 120. These results suggest strong coupling among hand joints during motion, emphasizing the dominance of full-skeleton motion in hand poses. Our method leverages this understanding, balancing long-term motion patterns and short-term articulation changes by factorization.

with respect to the entire skeleton, resulting in one or more relatively static joints. The change in the locations of moving joints with respect to such static joints can provide useful action cues. However, hand motions typically feature no such static reference points, as all the hand joints move together most of the time, making small changes in articulation to perform an action. To illustrate this difference, we randomly sample 1000 pose sequences from NTU RGB+D 120 [7] (full-body) and Assembly101 [10] (hands). We take the distances covered between two consecutive frames for each joint to form a distance array for the corresponding joint j in a given pose sequence, which is determined by-

$$d_j(t) = \left\| P_j(t) - P_j(t-1) \right\| \tag{1}$$

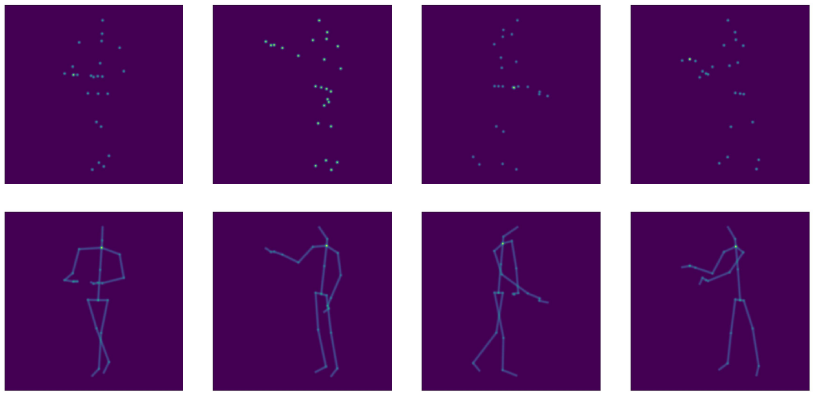
Here, $d_j(t)$ is the distance covered by joint j at frame t in reference to the previous frame, $P_j(t)$ and $P_j(t-1)$ are the 3D pose coordinates for joint j at time t and $t-1$, respectively. $\|.\|$ represents the Euclidean distance. Based on the sum of distances D_j for each joint j, we define the ***least active joint*** (static) and the ***most active joint*** (dynamic) for a particular sequence using the following equations-

$$\begin{gathered} D_j = \sum_{t=1}^{T} d_j(t) \\ j_{sta} = \arg\min_j D_j \qquad j_{dyn} = \arg\max_j D_j \end{gathered} \tag{2}$$

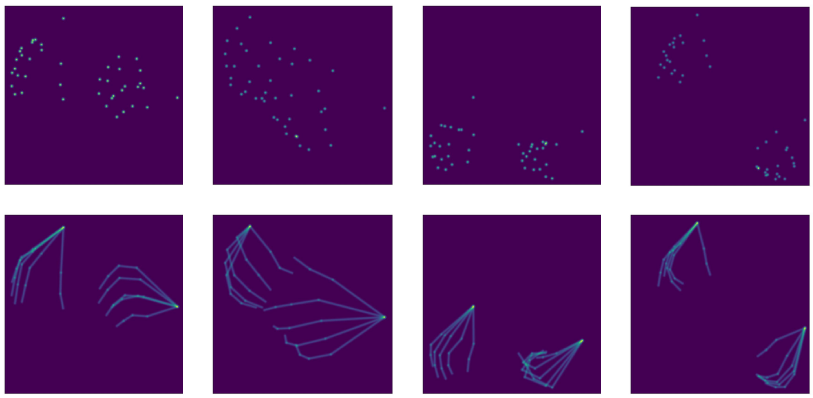
For each of the selected 1000 sequences, we take the two temporal sequences $\{d_{j_{sta}}(t)\}_{t=1}^{T}$ and $\{d_{j_{dyn}}(t)\}_{t=1}^{T}$, normalize the distance values using the diameters of the corresponding skeletons, and plot the sequences separately in Fig. 1. As can be observed, compared to the distances covered by the most active joints in NTU RGB+D 120, the least active joints show significantly lower movement, effectively serving as the static reference points. On the other hand, the distinction in distance arrays between the most and the least active joints in Assembly101 is less pronounced. In addition, we calculated the Pearson correlation coefficient, denoted as r, between $\{d_{j_{sta}}(t)\}_{t=1}^{T}$ and $\{d_{j_{dyn}}(t)\}_{t=1}^{T}$ for all Assembly101 sequences, resulting in a value of **0.93**. Conversely, for NTU RGB+D 120, the corresponding correlation coefficient is **0.33**. This suggests strong coupling among hand joints during motion, and full-skeleton movement is more dominant in hand poses compared to full-body poses. Consequently, modeling dependencies between spatiotemporally distant joints is less effective for the highly dynamic hand motion (also discussed in Sec. 3). Therefore, by considering both long-term motion patterns and short-term articulation changes, our method facilitates efficient spatiotemporal factorization through micro-actions (Sec. 4.1). We also incorporate the full-skeletal motion from the entire action during micro-action encoding, using a global wrist token as a reference (Sec. 4.2).

B 2D vs. 3D Pose for Hand Actions

For skeleton-based action recognition, PoseConv3D [2] proposes using 2D poses as input, arguing that the quality of pose estimation is superior in 2D. By constructing 3D heatmap volumes from 2D poses and employing a simple 3D-CNN, they surpass state-of-the-art GCN-based methods that rely on 3D poses. Incorporating CNN-based modeling for the pose stream also facilitates seamless integration with the RGB modality. In this section, we assess this proposition specifically within the context of hand skeletons.



(a) NTU RGB+D 120 [7].



(b) Assembly101 [10].

Fig. 2: Heatmaps for joints and limbs for (a) full-body poses and (b) hand poses.

Fig. 2 illustrates sample heatmaps from NTU RGB+D 120 [7] and Assembly101 [10]. Keypoints in full-body human poses are often prominently situated, with minimal self-occlusion, and the subject is typically centered within

Method	Input Pose	Verb Accuracy (%)
PoseConv3D [2]	2D	46.71
HandFormer-B/6	2D	58.92
HandFormer-B/6	3D	63.70

Table 1: Impact of using 2D vs. 3D poses as input for skeleton-based action recognition in hands. Experiments are done for verb recognition on Assembly101 [10].

the frame. As viewed in Fig. 2a, reducing pose dimensions to 2D does not significantly compromise detail; rather, it enhances input reliability by simplifying the pose estimation problem. However, this advantage diminishes when applied to hand poses. Hand poses present unique challenges, such as frequent self-occlusion and closer proximity of the keypoints, which are exacerbated by reducing the dimension to 2D. To empirically analyze this phenomenon, we evaluate HandFormer-B/6 with 2D and 3D poses for recognizing verbs in Assembly101 [10] and report in Tab. 1. This analysis reveals about 5% difference in favor of the 3D input. Furthermore, PoseConv3D [2] introduces a CNN-based approach with 2D keypoints, which directly utilizes heatmaps from the pose estimator or generates Gaussian heatmaps from the 2D coordinates. However, feeding heatmaps can diminish the clarity of keypoints to the model, particularly when they are in close proximity, as is often the case with hand poses. Hence, PoseConv3D [2] performs poorly in recognizing hand actions, as evident in Tab. 1.

In summary, although skeleton-based methods represent a broader field for action recognition with poses, they often lack the necessary adaptation for directly addressing hand-specific actions. This demands dedicated research on hand poses for hand-object interaction understanding.

C Alternatives for Frame Encoder

We utilize pre-trained ViT-g/14 and ViT-L/14 models from DINOv2 [9] followed by a linear layer without any fine-tuning as the frame encoder F for As-

Method Variant	Frame Encoder	Action	Verb	Object
RGB-only	ViT-g/14	32.07	55.61	44.89
	ResNet50	35.09	56.59	48.54
Pose+RGB	ViT-g/14	41.06	69.23	51.17
	ResNet50	41.99	69.28	51.96

Table 2: Comparison of different frame encoder options on Assembly101 [10]. Frame-wise TSM features from pretrained ResNet50 perform better compared to all-purpose features generated by DINOv2 with a ViT-g/14 backbone. RGB-only variant greatly benefits from the pretraining as it works with domain-specific features for action recognition. However, incorporating complementary pose modality reduces the gain from pretraining.

sembly101 [10] and H2O [5], respectively. Except for our experiments on Assembly101 [10] with monochrome egocentric videos (Sec. 5.5), all our multimodal results are obtained using DINOv2 features. This approach enables us to assess the effectiveness of image-based foundation models in videos, leveraging all-purpose features from RGB frames and achieving strong cross-view generalization performance (Sec 5.4). Although this is our default choice offering easy adaptation to new datasets with faster training, it can be considered compute-heavy when deployed in a low-resource setting due to the large ViT backbones. For better efficiency during inference, we propose a pretraining scheme that allows us to use a ResNet50 [4] that replaces the ViT without compromising accuracies, as shown in Tab. 2.

Specifically, we first train a TSM [6] model with a ResNet50 [4] backbone for action recognition, utilizing all action clips and then dropping the classification layer. This ResNet50 becomes the frozen image encoder in our proposed architecture, replacing ViT. During the training and inference of HandFormer, the TSM backbone operates as a true image model (ResNet50), as we employ it on individual frames without any channel shifting. The TSM features provided in the Assembly101 [10] are generated in this way, and we utilize them in our egocentric action recognition experiments (Sec. 5.5).

In Tab. 2, we present a comparison of the two backbone options for our frame encoder – ResNet50 from TSM and ViT-g/14 from DINOv2. The ResNet50 outputs, enhanced through pretraining within TSM, incorporate domain-specific features and temporal encoding via channel shifting during training. As a result, the RGB-only variant achieves a 3% higher action accuracy compared to using DINOv2 features alone. However, when introducing pose information, the image-based features are complemented by motion features, reducing the impact of motion understanding facilitated by the temporal shift mechanism of TSM in the ResNet50 encoder. Therefore, integrating pose data diminishes the pretraining advantage of ResNet50, resulting in a performance gap of less than 1%.

D Maintaining High Temporal Resolution at Low Cost

Our method is designed to perform action recognition efficiently in hand-object interaction videos. Obtaining efficiency in such a setup is challenging as we need to maintain a high temporal resolution to understand fine-grained hand movements that constitute the actions. Therefore, we propose HandFormer using densely sampled pose frames and sparse RGB frames. In this section, we quantify the efficiency of this method compared to an alternative video model. As mentioned, understanding fine-grained hand motion demands a high temporal resolution to differentiate verb classes. For instance, relying on sparsely sampled frames may make actions like "*screwing*" and "*unscrewing*" indistinguishable. However, adopting a high temporal resolution with video models operating on RGB frames is challenging, primarily due to *(i)* the excessive computation associated with performing spatiotemporal operations on numerous frames, and *(ii)*

Method	Component	GFLOPs	Count	Total GFLOPs
TSM [6]	-	-	-	669.79
HandFormer-B/21	Pose Estimator [3]	0.30	162	84.01
	Frame Encoder	4.12	8	
	Trajectory Encoder	0.29	8	
	Multimodal Tokenizer	0.01	8	
	Temporal Transformer	0.05	1	

Table 3: Comparison of FLOPs between HandFormer and TSM [6] when both maintain a high temporal resolution at 60 fps. The number of frames is determined by the average action duration in Assembly101 [10], and we use eight non-overlapping micro-actions in our model.

the need to address redundancy in RGB frames to extract meaningful information.

In Tab. 3, we compare the FLOPs of our model vs. an efficient video model, TSM [6] with a ResNet50 backbone when both maintain a high temporal resolution. The results reveal that our model operates at about 8× fewer FLOPs. As TSM has a 2D backbone and no 3D convolutions, it is expected to represent the lower bound for the computational cost of a video model at that temporal resolution. For our frame encoder, we opt for the efficient alternative as described in Sec. C. The average duration of fine-grained actions in Assembly101 [10] is 1.7 seconds. Following [10], we include an additional 0.5 seconds of context on either side, resulting in an average of $2.7 \times 60 = 162$ frames per action clip. We use $K = 8$ non-overlapping micro-actions, thus sampling 8 RGB frames and using the trajectory encoder eight times.

E Additional Details for Multimodal Training

Our training recipe for the multimodal HandFormer involves initializing the trajectory encoder with pretrained weights and utilizing hand-object ROI crop within the frame encoder — ensuring better use of pose and RGB, respectively.

E.1 Pretraining Trajectory Encoder

Encoding micro-action involves extracting RGB and pose features using frame encoder F and trajectory encoder T, respectively. While the frame encoder stays frozen and provides the appearance features, the trajectory encoder is learned and is expected to capture the hand motion. To effectively guide the trajectory encoder in achieving the desired encoding, we pretrain it for verb recognition solely using pose input. This approach leverages the inherent ability of pose data to capture hand motion, a key determinant of the verb while remaining agnostic to explicit information about interacting objects. This pretraining scheme leads to a better initialization of the trajectory encoder in multimodal HandFormer

Frame Encoder	Trajectory Encoder Pretraining	Accuracy(%)		
		Action	Verb	Object
ViT-g/14	✗	39.79	67.40	50.69
	✓	41.06	69.23	51.17
ResNet50	✗	40.47	66.00	51.10
	✓	41.99	69.28	51.96

Table 4: Initializing the trajectory encoder T with pretrained weights improves the overall performance with better verb recognition capability. Results are on Assembly101 [10] dataset. The initial weights for T are obtained by training the model to predict the verb classes from pose-only input.

for action recognition. In Tab. 4, we observe that initializing the trajectory encoder with pretrained weights leads to improved action recognition performance, particularly enhancing the recognition of verb classes.

E.2 Hand-Object Interaction Crop

In hand-object interaction (HOI) videos, the region of interest typically centers around the hands, capturing crucial information about the interacting object and the type of interaction. Leveraging 3D hand poses obtained through a readily available pose estimator [3], we project these poses onto RGB frames, extract the enclosing rectangle of the projected 2D pose, and expand it by 25% to define the ROI crop. However, relying solely on the cropped region can occasionally mislead the model for three potential reasons: *i)* failure of the pose estimator on certain frames, leading to the absence of useful features from the RGB frames, *ii)* the full object might not be visible when the crop is taken based on hand poses only, and *iii)* hand crops have limitations in capturing global changes compared to the full frames. Hence, to capitalize on both the localized interaction information of hand crops and the global contextual information provided by full frames, our model combines them both. If a valid hand crop is found, we take the full and cropped RGB frames, pass them through the frame encoder, average their features, and re-normalize them to unit norm. This full vs. HOI crop ablation is shown in Tab. 5, in which combining both performs better than the alternatives.

Full Frame	HOI Crop	Accuracy(%)		
		Action	Verb	Object
✓	✗	38.73	68.31	48.77
✗	✓	38.44	68.95	48.20
✓	✓	41.06	69.23	51.17

Table 5: Ablation study comparing full vs. HOI cropped RGB frames on Assembly101 [10]. Incorporating both full and cropped RGB frames allows for leveraging localized interaction details from hand crops and global contextual information from full frames, resulting in improved accuracy. HandFormer-B/21 is used with eight non-overlapping micro-actions.

F Efficiency Comparison with Shift-GCN

While MS-G3D [8] and ISTA-Net [12] show state-of-the-art performance for action recognition with hand poses, they are not efficiency-focused. Our HandFormer outperforms them with significantly fewer FLOPs. However, HandFormer-B/6 prioritizes efficiency while slightly trading off accuracy. Therefore, we implement and test an efficiency-focused baseline, ShiftGCN [1], for verb recognition on Assembly101 [10] and compare it to HandFormer-B/6 in Tab. 6. While Shift-GCN relies on graph shift operations and pointwise convolutions for efficiency, our model identifies the crucial joints, *i.e.*, the fingertips and the wrist joint, and processes only these joints to reduce FLOPs substantially. As evident from Tab. 6, our model outperforms Shift-GCN while incurring lower FLOPs.

Method	**GFLOPs**	**Verb Accuracy (%)**
Shift-GCN [1]	2.11	63.14
HandFormer-B/6	1.33	63.70

Table 6: Comparison of HandFormer-B/6 with Shift-GCN, an efficiency-focused baseline for skeleton-based action recognition. Experiments are done for verb recognition on Assembly101 [10].

G Qualitative Analysis

In this section, we analyze the class-wise verb accuracy using the pose-only HandFormer, aiming to identify the model's limitations. Furthermore, we examine the multimodal aspect of action recognition and its role in alleviating object misclassification.

G.1 Pose-only Performance

Fig. 3 displays the confusion matrix for verb classes using HandFormer-L/21 on the test set. Notably, *inspect*, *rotate*, *position*, and *remove* verbs present recognition challenges despite ample dataset samples. One potential explanation for this phenomenon is the shared presence of certain signature movements among these classes, which also occur in two head classes, namely, *pick up* and *put down*. Another interesting observation in the results is the frequent classification of *'attempt to x'* classes as *'x'*. This is expected, as determining the successful completion of a task adds another layer of complexity to these classes, especially when relying solely on pose information without considering changes in the appearance of the interacting object throughout the clip.

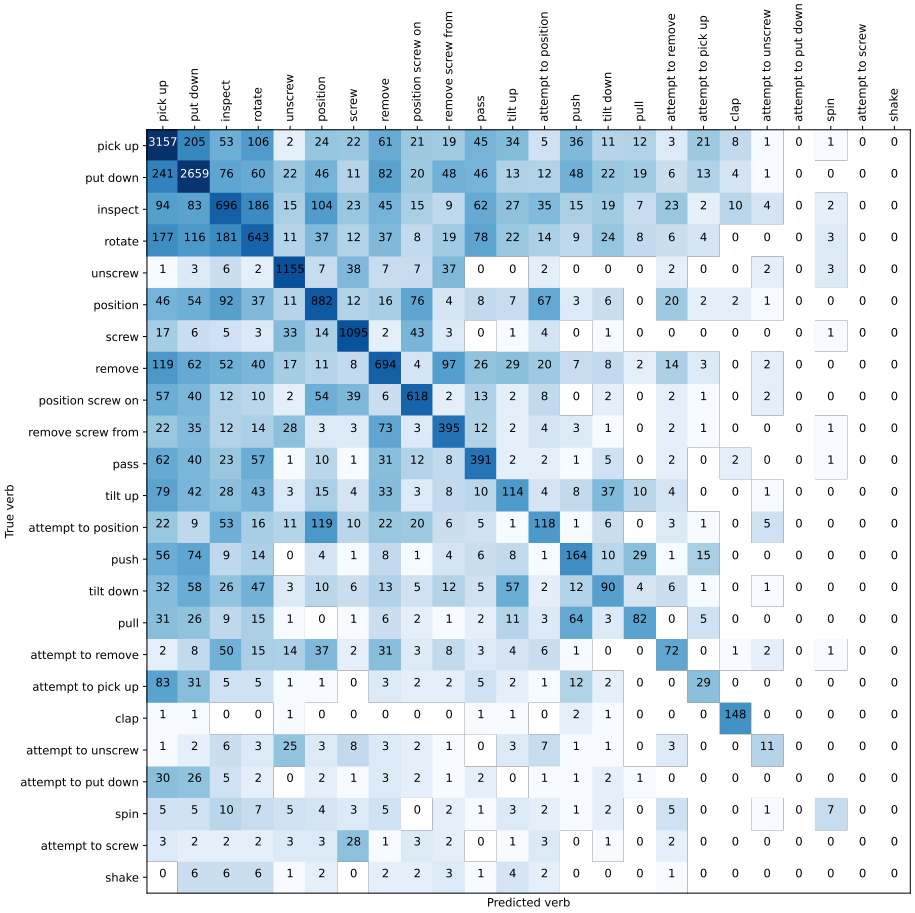


Fig. 3: Confusion matrix for pose-only verb recognition with HandFormer-L/21.

G.2 Multimodal Fusion

To gain insights into how appearance information from RGB complements pose-based models in hand-object interaction scenarios, we analyze samples involving *put down* actions. In Tab. 7, we showcase the action classes predicted for these samples using our pose-only model, referred to as *Pose + 0 RGB*. In these samples, the model successfully detected the verb but struggled with object classification. This challenge arises due to similarities in articulations observed during tasks such as grasping a screwdriver and a screw or differentiating between a partially assembled toy and a completed one. These similarities lead to misclassifications by the pose-only model. However, introducing a single RGB frame, denoted as *Pose + 1 RGB*, enhances the model's ability to correctly identify the relevant object by providing visual context. This observation highlights

Pose + 0 RGB	Put down screwdriver	Put down screw	Put down partial toy
RGB Sample			
Pose + 1 RGB	Put down screw	Put down screwdriver	Put down finished toy

Table 7: Action classification by our model with and without sampling an RGB frame. Incorrect predictions are highlighted in red, while correct predictions are marked in green.

the limitations of recognizing actions, *i.e.* verb+object, solely from hand poses, emphasizing the importance of incorporating visual cues.

References

1. Cheng, K., Zhang, Y., He, X., Chen, W., Cheng, J., Lu, H.: Skeleton-based action recognition with shift graph convolutional network. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 183–192 (2020)
2. Duan, H., Zhao, Y., Chen, K., Lin, D., Dai, B.: Revisiting skeleton-based action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2969–2978 (2022)
3. Han, S., Liu, B., Cabezas, R., Twigg, C.D., Zhang, P., Petkau, J., Yu, T.H., Tai, C.J., Akbay, M., Wang, Z., et al.: Megatrack: monochrome egocentric articulated hand-tracking for virtual reality. ACM Transactions on Graphics (ToG) **39**(4), 87–1 (2020)
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
5. Kwon, T., Tekin, B., Stühmer, J., Bogo, F., Pollefeys, M.: H2o: Two hands manipulating objects for first person interaction recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10138–10148 (2021)
6. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7083–7093 (2019)
7. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C.: 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 (2019)
8. Liu, Z., Zhang, H., Chen, Z., Wang, Z., Ouyang, W.: 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 (2020)

9. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
10. Sener, F., Chatterjee, D., Shelepov, D., He, K., Singhania, D., Wang, R., Yao, A.: Assembly101: A large-scale multi-view video dataset for understanding procedural activities. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21096–21106 (2022)
11. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: 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)
12. Wen, Y., Tang, Z., Pang, Y., Ding, B., Liu, M.: Interactive spatiotemporal token attention network for skeleton-based general interactive action recognition. arXiv preprint arXiv:2307.07469 (2023)