Skarimva: skeleton-based action recognition is a multi-view application

Daniel Bermuth, Alexander Poeppel, Wolfgang Reif

Angaben zur Veröffentlichung / Publication details:

Bermuth, Daniel, Alexander Poeppel, and Wolfgang Reif. 2026. “Skarimva: skeleton-based action recognition is a multi-view application.” In *2026 IEEE 20th International Conference on Automatic Face and Gesture Recognition (FG), May 25-29, 2026, Kyoto, Japan*, 1–5. Piscataway, NJ: IEEE. https://doi.org/10.1109/FG67764.2026.11556972.

Skarimva: Skeleton-based Action Recognition is a Multi-view Application

Daniel Bermuth, Alexander Poeppel, Wolfgang Reif
Institute for Software and Systems Engineering, University of Augsburg, Germany
daniel.bermuth@uni-a.de

***Abstract*— Human action recognition plays an important role when developing intelligent interactions between humans and machines. While there is a lot of active research on improving the machine learning algorithms for skeleton-based action recognition, not much attention has been given to the quality of the input skeleton data itself. This work demonstrates that by making use of multiple camera views to triangulate more accurate 3D skeletons, the performance of state-of-the-art action recognition models can be improved significantly. This suggests that the quality of the input data is currently a limiting factor for the performance of these models. Based on these results, it is argued that the cost-benefit ratio of using multiple cameras is very favorable in most practical use-cases, therefore future research in skeleton-based action recognition should consider multi-view applications as the standard setup.**

I. Introduction

To allow a computer system to intelligently react to human actions, it is important that it can recognize these actions reliably. Skeleton-based action recognition has become a popular approach for this task, because the skeleton data is compact, robust to changes in human appearance and the surroundings, and still preserves the relevant motion information.

However, most research in this field has focused on improving the machine learning algorithms, while comparatively less attention has been given to the input data itself. In fact, progress in recent years appears to have plateaued, with only incremental accuracy gains being reported, despite the use of increasingly complex models. This work demonstrates that by rethinking the input data acquisition process, significant accuracy improvements can be achieved with existing models. For example, the error rate on the widely used *NTU-RGBD-60* dataset could be reduced by over 50% across different backbones, achieving new state-of-the-art results under standard evaluation protocols. Therefore, the input data quality seems to have been a limiting factor for the performance of current skeleton-based action recognition models.

Following these results, it is proposed to consider skeleton-based action recognition as a multi-view application, instead of the current single-view standard. While using multiple cameras increases the system complexity slightly, the setup of additional cameras is relatively simple and worth the small extra effort in most practical applications, as will be discussed later in this work.

To support future research, the complete code is open-sourced at: `https://gitlab.com/Percipiote/`

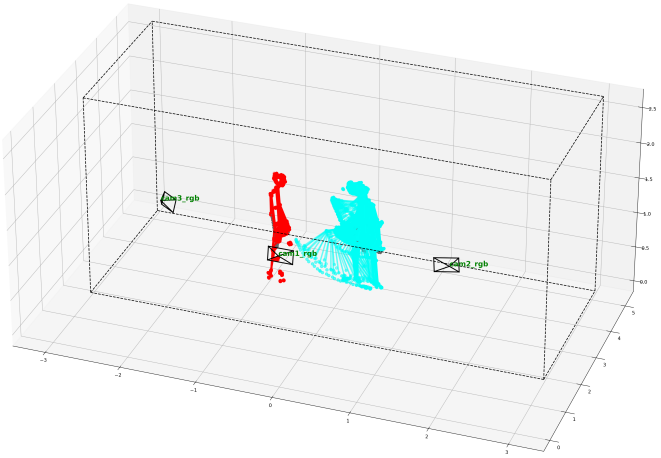

Fig. 1: Example of a *kick other person* action with the new multi-view whole-body skeletons.

II. Related Work

Only a few works have investigated the influence of input skeleton quality so far, which will be the focus in the following. Since the amount of action-recognition datasets with multi-view image sequences, which are required for this investigation, is very limited, the following experiments are based on the *NTU-RGBD* dataset [18], [26], which is also the only one of this type that is widely used in the field. Being one of the most popular action-recognition datasets has furthermore the benefit of allowing an experimental comparison with many previous works.

The original *NTU-RGBD* dataset [18], [26] was created with three non-calibrated *Kinect* RGB-D cameras, from which each camera's 3D skeleton estimation was processed independently. While the skeleton quality was sufficient at the time of release, it still contains a notable amount of poor joint estimations, caused by occlusions, depth ambiguities, and the limited accuracy of the pose estimation algorithms available at the time. The authors of *PoseConv3D* [11] found that using 2D poses from a more recent pose estimator already improves the action recognition accuracy notably, despite discarding the 3D distance information. In *NTU-X* [27], a different approach was followed, in which new whole-body 2D and 3D skeletons were created with a single-view 2D-to-3D lifting approach. This also resulted in accuracy improvements.

Both methods, though, are limited by the single-view setting, which is susceptible to depth ambiguities and occlusion problems. In contrast, this work explores a multi-view triangulation approach to even further improve the skeleton quality, which leads to significantly better action recognition

results. A more detailed comparison with these previous works is provided later in the experiments section.

III. Improving Pose Estimates

One of the main problems that prevents directly using a multi-view approach on the *NTU-RGBD* dataset is that the camera calibrations and synchronizations are not available. Therefore, a reconstruction procedure was developed to recover intrinsic and extrinsic camera parameters by minimizing multi-view skeleton reprojection and pose alignment errors (using the original camera-aligned skeletons) with iterative outlier removal, as well as to estimate temporal alignment between the video streams. Additional details about this procedure are provided in the appendix. The source-code was integrated into the *skelda* library [2].

To calculate new skeleton poses, the state-of-the-art 3D multi-view multi-person pose estimator *RapidPoseTriangulation* [3] can now be used. It was shown to generalize well between different camera setups and datasets, and is one of the very few which also supports whole-body pose estimation with face and finger keypoints. It also has the benefit that it does not require re-training for each camera setup, which is often the case in learned multi-view approaches [3]. The pose estimation, with about 130 FPS on an Nvidia RTX4080, runs faster than the camera frame-rate of 30 FPS.

To create motion sequences again, the skeletons are matched to persons over time using a simple dataset-optimized tracking approach. Specifically, a new skeleton is assigned to an existing track if the average joint distance is below a certain threshold. If no matching track is found, a new track is initialized. After all skeletons have been assigned to tracks, short tracks below a certain length are removed as false detections. If two tracks do not overlap in time, they are merged into one. Since this dataset has at most two persons in a scene, only the two longest tracks are kept and additional tracks are discarded.

IV. Results

To evaluate the benefits of the new multi-view skeletons, three recent skeleton-based action recognition models were trained and evaluated on the *NTU-RGBD-60* [26] and *NTU-RGBD-120* [18] datasets. The models used are *MSG3D* [19], *DG-STGCN* [9], and *ProtoGCN* [16], which are all graph convolutional network (GCN) based models that have shown strong performance on these datasets. For the first two models, the implementations from *PySKL* [10] were used. In general, only a few modifications were necessary to train the models with the new skeletons, mostly related to the different joint counts and types. The training hyperparameters were kept identical to the original implementations as far as possible, to ensure comparability. As augmentations, only random rotations around the z-axis (which are now easy with the world-aligned skeletons) and size scaling were used.

A. Improvements with new whole-body poses

Table I demonstrates that using the new multi-view whole-body skeletons leads to significant accuracy improvements for all three models on both dataset splits. Note that in contrast to the commonly used multi-stream ensembling approach, in this experiment all input modalities (*j+b+jm+bm*) are concatenated and fed into a single model, which reduces the training effort notably. Further, only a single sequence is sampled for each test sequence to speed up testing.

Method	original	wb25	wb137	wb69	wb31	body
MSG3D	90.8	96.5	94.0	96.3	96.4	94.0
DG-STGCN	92.0	96.7	94.4	96.5	96.9	94.6
ProtoGCN	91.8	97.1	94.3	95.6	96.6	95.5
MSG3D	87.0	93.6	90.4	93.3	93.5	88.2
DG-STGCN	88.3	94.6	92.7	94.9	94.1	89.7
ProtoGCN	88.7	94.8	91.6	94.4	94.3	89.9

TABLE I: Results with the new whole-body poses on *NTU-RGBD-60* and *NTU-RGBD-120*. The split *wb25* has the same keypoints as the original one, *wb137* has all whole-body keypoints, *wb69* has no additional face keypoints, and *wb31* only adds the other three fingertips.

An interesting observation is that adding more finger and face keypoints does not necessarily improve the accuracy further. This could be an indication that the models start to overfit on keypoints that are not particularly relevant for the action classes. This effect was also observed in *NTU-X* [27]. Additionally, the models become significantly slower with more input joints due to the strong increase in graph edges. These findings raise the question for future work of how the models can be designed to use the whole-body keypoints more efficiently and focus more on the relevant keypoints automatically themselves.

B. Comparison with other skeleton recalculations

Besides the approach followed in this work, two other methods for improving the skeletons of *NTU-RGBD* have been evaluated already. As described previously, *PoseConv3D* [11] has introduced higher-quality 2D body-only skeletons from a more recent 2D estimator, and *NTU-X* [27] new whole-body skeletons with a single-view 2D-to-3D lifting approach. The results in Table II show that the multi-view triangulation approach proposed in this work outperforms both previous methods by a significant margin.

Method	body	wb25	body+fingers	body+hand+face
original [19]	-	91.5	-	-
PoseConv3D [11]	92.5	-	-	-
NTU60-X [27]	91.3	-	91.8	91.1
Skarimva	**94.0**	**96.5**	**96.3**	**94.0**
original [19]	-	86.9	-	-
PoseConv3D [11]	87.2	-	-	-
NTU120-X [27]	84.5	-	87.1	-
Skarimva	**88.2**	**93.6**	**93.3**	**90.4**

TABLE II: Comparison with other skeleton representations using the *MSG3D* model. Note that there are small differences in the joint types and ensembling methods, but they are not expected to significantly impact the results, as explained in more detail in the text.

Note that there exist some slight training differences, but they are unlikely to significantly affect the results. The *NTU-X body* set includes 6 foot keypoints, whereas the one from *PoseConv3D* and *Skarimva* only use the standard *COCO* [14] body joints. The *body+fingers* set is basically the same, except that *Skarimva* uses 2 additional joints derived from the body joints. The face joints of *NTU-X* do not contain keypoints for the cheeks, so their *body+fingers+face* set has 118 joints instead of the 137 in *Skarimva*. Regarding model ensembling, the original results of *MSG3D* use a two-stream approach with joint and bone input modalities (*j+b*), in which each output of the two models is ensembled by averaging the two prediction distributions. The papers of *NTU-X* [27] and *PoseConv3D* [11] do not explain their ensembling method, but from the source-code it seems they use the same. The results with *Skarimva* do not use ensembling, but a single model with all four input modalities (*j+b+jm+bm*), as in Table I. This way only one model has to be trained, which notably reduces training effort, and as shown in Table VIII and IX in the appendix, the accuracy stays on a similar level.

While the results of *PoseConv3D* [11] with the better body-only skeletons show that improving the quality of the 2D pose estimates already is beneficial, the authors also noticed a larger drop in performance when projecting the original 3D poses back to 2D ([11], Table 16), caused by the loss of helpful 3D information. This can also be seen when comparing the accuracy to the new 3D body-only poses of this work, which is clearly higher (while both methods used different 2D pose estimators, their performance is on a similar level, so the influence from this should be minimal).

The results of *NTU-X* [27] show that adding fingers to the skeletons is important to improve the average accuracy, but their lifting-based 3D reconstruction method is notably inferior to the multi-view approach proposed in this work. Another experiment in *PoseConv3D* also showed that lifting itself does not help ([11], Table 13). An explanation for this effect could be that the action recognition model already receives the pose input over time, so its input is similar to that of the lifting models. Therefore, if doing so would be helpful, the action model could learn to perform the lifting internally. Under this assumption an additional lifting model would not add any new information, but instead could introduce additional errors from the lifting process. The only benefit of an external lifting model could be that it can be pre-trained with general human motions on a larger dataset, because no action labels are required. However, such a pre-training would be possible for the multi-view pose estimation approaches as well. In fact, most of the lifting datasets are created from multi-view datasets already. In theory, depth triangulation is possible in a single-viewpoint video over time as well, using invariant distances (like limb lengths) moving in front of the camera, but multi-view triangulation is much easier and more computationally efficient. Using multiple views also has the important advantage that (self-) occlusions can be partially avoided, whereas a video model would need to infer occluded keypoints from context, which is susceptible to errors.

C. With multi-stream ensembling

The common evaluation procedure of previous works uses multi-stream ensembling of multiple input modalities, combined with creating multiple randomized samples from a single test sequence, to improve the accuracy further. Note that since this procedure is computationally very intensive, it is not suited for real-time applications. The results with this evaluation method are given in Table III for completeness, and set new state-of-the-art scores. The model performance on the single modalities is provided in the appendix.

Method		xsub
MSG3D [19]	'2020	91.5
CTR-GCN [6]	'2021	92.4
ST-GCN++ [10]	'2022	92.6
MotionBert [39]	'2023	93.0
InfoGCN [7]	'2022	93.0
BlockGCN [38]	'2024	93.1
MMP-ST [36]	'2025	93.1
LGS-Net [23]	'2025	93.2
DG-STGCN [9]	'2022	93.2
HD-GCN [13]	'2023	93.4
SkateFormer [8]	'2024	93.5
LA-GCN [32]	'2023	93.5
TDSN-GCN [15]	'2025	93.6
MSA-GCN [1]	'2024	93.6
DE-GCN [22]	'2024	93.6
JMDA [31]	'2025	93.7
Hyper-GCN [37]	'2025	93.7
Shap-Mix [34]	'2024	93.7
ProtoGCN [16]	'2025	93.8
PoseC3D [11]	'2022	94.1
Hulk [30]	'2025	94.3
SkeletonAgent [17]	'2025	94.5
3Mformer [28]	'2023	94.8
LLM-AR [24]	'2024	95.0
POTR [5]	'2025	95.3
ProtoGCN+Skarimva		**97.5**

Method	xsub
MSG3D [19]	86.9
PoseC3D [11]	86.9
ST-GCN++ [10]	88.6
LLM-AR [24]	88.7
CTR-GCN [6]	88.9
DG-STGCN [9]	89.6
SkateFormer [8]	89.8
InfoGCN [7]	89.8
HD-GCN [13]	90.1
MMP-ST [36]	90.2
BlockGCN [38]	90.3
Shap-Mix [34]	90.4
MSA-GCN [1]	90.6
LA-GCN [32]	90.7
JMDA [31]	90.9
Hyper-GCN [37]	90.9
ProtoGCN [16]	90.9
DE-GCN [22]	91.0
TDSN-GCN [15]	91.1
POTR [5]	91.1
SkeletonAgent [17]	91.7
3Mformer [28]	92.0
ProtoGCN+Skarimva	**95.4**

TABLE III: Results on *NTU-RGBD-60* and *NTU-RGBD-120*.

D. Few-shot learning

In real-world applications, it is often unlikely to have large amounts of labeled training data (here around 200 training examples per class), because labeling is time-consuming and costly. Therefore, the ability to classify new actions with only a few examples is an important feature for action recognition models. Thus, the following experiments evaluate few-shot learning capabilities with the new skeleton poses.

Because the source-code of the top three previous works was either incomplete or not available, the currently used classification models were extended instead. In particular, their default linear-layer classification head was replaced by a combined contrastive embedding and classification head, and the data sampling process was modified to support the new training scenario. The idea of the embeddings is to map samples of the same class closer together in the embedding space, while pushing samples of different classes further apart. In evaluation, when new classes are introduced, their embeddings can be calculated from a single example, and then new samples can be classified by calculating the shortest distance to the example embeddings.

Table IV shows that the current state-of-the-art performance for one-shot learning can be improved notably with the new skeletons as well.

Method	120
APSR [18]	45.3
TCN OneShot [25]	46.5
SL-DML [21]	49.6
Skeleton-DML [20]	54.2
MotionBert [39]	61.0
Koopman [29]	68.1
M+C-scale [33]	68.7
SkeletonX [35]	69.1
ProtoGCNce+Skarimva	**76.0**

TABLE IV: One-shot transfer classification on *NTU-RGBD*.

Using only one labeled example is considered unreasonable though, because in practical applications it should be possible to provide a few more examples with low effort, especially if accuracy benefits from it. Therefore, Table V investigates the results with five examples per action class, a scenario that leads to notably better performance.

Method	120
nearest neighbor	84.7
5-nearest neighbor	84.9
prototype matching	84.9

TABLE V: Few-shot transfer with five examples per action.

E. Real-time Applications

Real-time capability is an important factor for many applications. However, as it is outside the scope of this section and unrelated to the new skeleton poses, some further notes on this topic are provided in the appendix.

V. Discussion

The title of this work frames skeleton-based action recognition as a multi-view problem, which is discussed in more detail in this section.

The experimental results clearly showed that improving the 3D skeleton quality by multi-view triangulation has led to significant performance gains for state-of-the-art skeleton-based action recognition models. This suggests that the quality of the input data was a limiting factor for the performance of these models.

Using a multi-view approach is a straightforward way to improve the 3D pose quality, as depth ambiguities and occlusions can be reduced by observing the subject from different angles. A similar principle underlies binocular vision in animals, where multiple viewpoints improve depth perception and robustness.

From a practical perspective, the main drawback of a multi-view setup is the need for additional cameras. However, the added system complexity is often moderate in relevant application scenarios.

In professional setups, like sports analytics, surveillance, or robotics, the effort of installing another camera should be negligible compared to the overall system setup, and often multiple cameras are used already for other purposes anyway. In consumer applications, those in which reliable recognition accuracy has some importance, a simple multi-camera setup can be realized as well with limited effort. Basically, non-experienced users could buy two or three inexpensive USB cameras, plug them into a computer, and mount them in a static position. Then they could calibrate them by printing out a chessboard pattern, or displaying one on their smartphone screens, and moving it around in front of the cameras, while they are guided by a software tool. For mobile applications, like action recognition on smartphones, many devices already have multiple cameras built-in nowadays, which could be used for multi-view triangulation if their relative distance is large enough, otherwise an external clip-on camera could be added instead.

While accurate camera calibration and the synchronization of their image streams is beneficial, especially in professional setups, it is not strictly necessary. The observed gains in this work were achieved with rather rough camera calibrations and without synchronization at all. So an inexpensive home setup as explained before will be able to benefit from the additional views as well.

Regarding the additional computational cost, because now multiple image streams have to be processed, this is not a big issue either. As mentioned earlier, *RapidPoseTriangulation* can process multiple images on consumer-grade hardware in real-time already, and faster than most consumer-grade cameras will provide them.

In summary, using multiple views for skeleton-based action recognition is a simple and effective way to notably improve the recognition accuracy, offering a favorable cost-benefit ratio in many practical applications.

VI. Conclusion

This work has shown that improving the input data quality is an effective way to boost the performance of skeleton-based action recognition models. In particular, it has been demonstrated that leveraging multiple camera views to triangulate 3D skeletons leads to substantially better skeleton poses, and in the following, to large accuracy gains. Across multiple settings, the accuracy error could be reduced by over 50% compared to the original skeletons, achieving new state-of-the-art results on the popular *NTU-RGBD* dataset.

Based on these findings, it can be concluded that skeleton-based action recognition should be a multi-view application, even though most research in this field uses single-view skeletons so far. Since the cost-benefit ratio of using multiple cameras is very favorable in many scenarios, it is recommended to use multiple cameras if possible for practical applications.

To support future research, for example towards even better camera calibration, or a more effective integration of all whole-body joints into the action recognition models, code and models used in this work are made publicly available.

REFERENCES

[1] K. C. Alowonou and J.-H. Han. MSA-GCN: Exploiting Multi-Scale Temporal Dynamics With Adaptive Graph Convolution for Skeleton-Based Action Recognition. *IEEE Access*, 2024.

[2] D. Bermuth, A. Poeppel, and W. Reif. VoxelKeypointFusion: Generalizable Multi-View Multi-Person Pose Estimation. *arXiv preprint arXiv:2410.18723*, 2024.

[3] D. Bermuth, A. Poeppel, and W. Reif. RapidPoseTriangulation: Multi-view Multi-person Whole-body Human Pose Triangulation in a Millisecond. *arXiv preprint arXiv:2503.21692*, 2025.

[4] D. Bermuth, A. Poeppel, and W. Reif. Tutabo-1: Towards Real-time Capable AI-based Safety Systems for Human-Robot Collaboration. In *2025 IEEE International Conference on Advanced Robotics (ICAR)*. Institute of Electrical and Electronics Engineers (IEEE), 2025.

[5] L. Cao, S. Huai, and J. Gai. Reenvisioning Skeleton-based Action Recognition Through the Lens of NLP. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[6] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, and W. Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13359–13368, 2021.

[7] H.-g. Chi, M. H. Ha, S. Chi, S. W. Lee, Q. Huang, and K. Ramani. Infogcn: Representation learning for human skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20186–20196, 2022.

[8] J. Do and M. Kim. Skateformer: skeletal-temporal transformer for human action recognition. In *European Conference on Computer Vision*, pages 401–420. Springer, 2024.

[9] H. Duan, J. Wang, K. Chen, and D. Lin. DG-STGCN: dynamic spatial-temporal modeling for skeleton-based action recognition. *arXiv preprint arXiv:2210.05895*, 2022.

[10] H. Duan, J. Wang, K. Chen, and D. Lin. Pyskl: Towards good practices for skeleton action recognition. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 7351–7354, 2022.

[11] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2969–2978, 2022.

[12] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *Proceedings of the IEEE international conference on computer vision*, pages 3334–3342, 2015.

[13] J. Lee, M. Lee, D. Lee, and S. Lee. Hierarchically decomposed graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10444–10453, 2023.

[14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[15] D. Liu, Y. Hu, K. Hua, Y. Lu, Z. Zhang, X. Ma, Z. Zhong, and P. Chen. TDSN-GCN: Transformerify Overall Structure Decaying Static Graph Embedding NAS-guided GCN for Skeleton Action Recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

[16] H. Liu, Y. Liu, M. Ren, H. Wang, Y. Wang, and Z. Sun. Revealing key details to see differences: A novel prototypical perspective for skeleton-based action recognition. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 29248–29257, 2025.

[17] H. Liu, Y. Liu, C. Wang, Y. Wang, and Z. Sun. SkeletonAgent: An Agentic Interaction Framework for Skeleton-based Action Recognition. *arXiv preprint arXiv:2511.22433*, 2025.

[18] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. 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, 2019.

[19] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 143–152, 2020.

[20] R. Memmesheimer, S. Häring, N. Theisen, and D. Paulus. Skeleton-dml: Deep metric learning for skeleton-based one-shot action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3702–3710, 2022.

[21] R. Memmesheimer, N. Theisen, and D. Paulus. Sl-dml: Signal level deep metric learning for multimodal one-shot action recognition. In *2020 25th International conference on pattern recognition (ICPR)*, pages 4573–4580. IEEE, 2021.

[22] W. Myung, N. Su, J.-H. Xue, and G. Wang. Degcn: Deformable graph convolutional networks for skeleton-based action recognition. *IEEE Transactions on Image Processing*, 33:2477–2490, 2024.

[23] Q. Pan and X. Xie. Language-guided temporal primitive modeling for skeleton-based action recognition. *Neurocomputing*, 613:128636, 2025.

[24] H. Qu, Y. Cai, and J. Liu. Llms are good action recognizers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18395–18406, 2024.

[25] A. Sabater, L. Santos, J. Santos-Victor, A. Bernardino, L. Montesano, and A. C. Murillo. One-shot action recognition in challenging therapy scenarios. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2777–2785, 2021.

[26] A. Shahroudy, J. Liu, T.-T. Ng, and G. 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*, pages 1010–1019, 2016.

[27] N. Trivedi, A. Thatipelli, and R. K. Sarvadevabhatla. NTU-X: an enhanced large-scale dataset for improving pose-based recognition of subtle human actions. In *Proceedings of the Twelfth Indian Conference on Computer Vision, Graphics and Image Processing*, pages 1–9, 2021.

[28] L. Wang and P. Koniusz. 3Mformer: Multi-order Multi-mode Transformer for Skeletal Action Recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5620–5631, 2023.

[29] X. Wang, X. Xu, and Y. Mu. Neural koopman pooling: Control-inspired temporal dynamics encoding for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10597–10607, 2023.

[30] Y. Wang, Y. Wu, W. He, X. Guo, F. Zhu, L. Bai, R. Zhao, J. Wu, T. He, W. Ouyang, et al. Hulk: A universal knowledge translator for human-centric tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[31] L. Xiang and Z. Wang. Joint mixing data augmentation for skeleton-based action recognition. *ACM Transactions on Multimedia Computing, Communications and Applications*, 21(4):1–24, 2025.

[32] H. Xu, Y. Gao, Z. Hui, J. Li, and X. Gao. Language Knowledge-Assisted Representation Learning for Skeleton-Based Action Recognition. *arXiv preprint arXiv:2305.12398*, 2023.

[33] S. Yang, J. Liu, S. Lu, E. M. Hwa, and A. C. Kot. One-shot action recognition via multi-scale spatial-temporal skeleton matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7):5149–5156, 2024.

[34] J. Zhang, L. Lin, and J. Liu. Shap-mix: Shapley value guided mixing for long-tailed skeleton based action recognition. *arXiv preprint arXiv:2407.12312*, 2024.

[35] Z. Zhang, W. Cai, Q. Liu, and Y. Wang. SkeletonX: Data-Efficient Skeleton-based Action Recognition via Cross-sample Feature Aggregation. *arXiv preprint arXiv:2504.11749*, 2025.

[36] L. Zhou and X. Jiao. Multi-modal and multi-part with skeletons and texts for action recognition. *Expert Systems with Applications*, page 126646, 2025.

[37] Y. Zhou, T. Xu, C. Wu, X. Wu, and J. Kittler. Adaptive hyper-graph convolution network for skeleton-based human action recognition with virtual connections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12648–12658, 2025.

[38] Y. Zhou, X. Yan, Z.-Q. Cheng, Y. Yan, Q. Dai, and X.-S. Hua. Blockgcn: Redefine topology awareness for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2049–2058, 2024.

[39] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, and Y. Wang. MotionBERT: A Unified Perspective on Learning Human Motion Representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15085–15099, 2023.

APPENDIX

A. Calibration Details

As mentioned in the main text, the calibration process can be split into three steps:

- Estimate the camera intrinsics
- Estimate the camera extrinsics
- Synchronize the image streams

To estimate the intrinsics, the provided original 2D and 3D skeletons can be used. The 3D skeletons are projected into the image plane using an initial guess for the intrinsic camera matrix and distortion coefficients. Then, the re-projection error between the projected 3D joints and the provided 2D joints is calculated and used to minimize the initial intrinsic parameters. To stabilize the optimization, the initial intrinsic parameters were calculated as an average of the *Kinect* cameras used in the *Panoptic* dataset [12], which uses the same camera model.

To estimate the extrinsics, a similar approach is taken. The provided 3D skeletons in camera coordinates should align with each other in world coordinates (except for the pose estimation errors). An initial guess for the extrinsic parameters is created by placing the cameras in a circle around the room origin, looking towards the center, using the height and distance mentioned in the dataset paper. The first central camera serves as static reference, and all camera-originated skeletons are transformed into world coordinates using the initial extrinsic parameters. Then an error is calculated between the first camera's skeleton and the other cameras' skeletons, which is used to optimize the extrinsic parameters of the two side-view cameras. After one round of optimization, 30% of the frames with the highest error are removed as outliers, and the optimization is repeated. This is done two times in total. To skip the additional matching complexity, only sequences with a single subject, and where the number of skeleton frames is equal for all three cameras, are used for the calibration process.

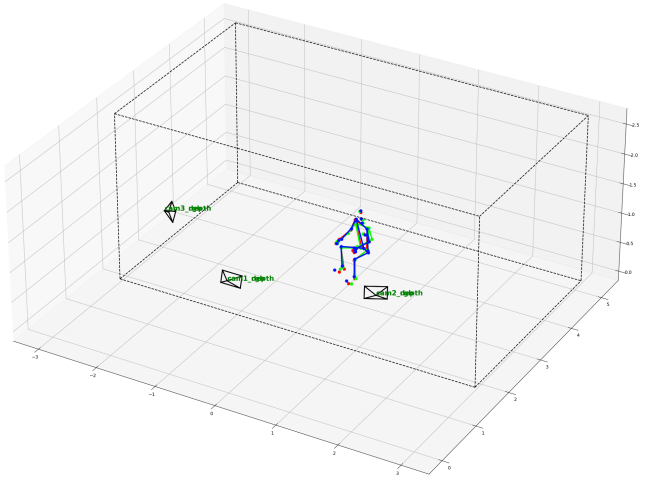

Fig. 2: Extrinsic calibration by overlapping skeletons from the different views. The overlap cannot be perfect due to errors in the original pose estimations.

To synchronize the images, a relatively simple approach is used. The original videos are split into image frames, and because the videos often do not have the same length and also slightly varying frame-rate, the average frame count is calculated, and the image frames of each camera are uniformly sampled by skipping or duplicating frames to match this average count. This way, the images are roughly aligned in time, although not perfectly synchronized.

A task for future work could be to further improve the calibration and synchronization, by also using image-space features, like objects in the background or the motion of the person in the different views. Such a method could then use the calibration parameters from this work to initialize their optimization algorithm. The current method calculates an average calibration for each setup and performer, but sometimes the cameras have slightly moved between different actions, or in the duration of one recording, and such movements are not taken into account yet, and can result in slight calibration offsets. But for now the results are already good enough to demonstrate the benefits of multi-view skeleton estimation.

B. Real-time Applications

Many applications require responding to human actions with only a small delay. Therefore, the prediction speed of the models is of interest as well. As can be seen in Table VI, the three tested models are rather small and can predict actions faster than most cameras deliver images, making them real-time capable.

Method	Size	FPS
MSG3D	3.14M	189
DG-STGCN	2.04M	140
ProtoGCN	4.46M	150

TABLE VI: Model size and inference speed. Tested on a single Nvidia-RTX4080 with a batch-size of 1.

By default, the input of the models is a complete action sequence. From those, a fixed number of images (here 100) is randomly sampled and fed into the model. For this approach the start and end of an action have to be known beforehand, but this is not very realistic for streamed camera inputs in a live setting. Therefore, to evaluate such a setting, the following experiment samples a fixed-length continuous sequence at a random starting position from the action sample instead. This would be similar to a windowed streaming approach. The results in Table VII show that this leads to a slight performance decrease, growing with smaller window sizes, but the accuracy is still reasonably high.

Method	3s	2s	1s
DG-STGCN	95.7	94.7	92.8
ProtoGCN	95.6	94.7	92.8

TABLE VII: Comparison of the effect of different input durations on *NTU-RGBD-60*, using a single random continuous subsequence of each sample.

C. Ensembling Modalities

For completeness, the performance results on the single input modalities are provided in the following. It can be seen that the combined *j+b+jm+bm* input is better than one modality alone, but if runtime is of no concern, training the models separately and fusing the outputs afterwards improves the results even more.

Joint	Bone	J-Motion	B-Motion	J+B+JM+BM
95.6	96.4	93.3	93.8	97.1

1J+1B-Fusion	2J+2B+1JM+1BM-Fusion
97.1	97.2

K	K-Motion	2J+2B+2K+1JM+1BM+1KM-Fusion
96.6	94.5	97.5

TABLE VIII: Comparison of input modalities and output fusions, using *ProtoGCN* on *NTU-RGBD-60-xsub*.

Joint	Bone	J-Motion	B-Motion	J+B+JM+BM
91.9	93.4	89.5	90.1	94.8

1J+1B-Fusion	2J+2B+1JM+1BM-Fusion
94.7	95.0

K	K-Motion	2J+2B+2K+1JM+1BM+1KM-Fusion
93.9	89.8	95.4

TABLE IX: Comparison of input modalities and output fusions, using *ProtoGCN* on *NTU-RGBD-120-xsub*.

Note that this evaluation concept reduces the runtime significantly, by $10\times$ for multi-sampling each test sequence and by another $6\times$ due to the different modalities. So the *ProtoGCN* model can only process around 3 instead of 150 samples per second, notably reducing its real-time capability.

Because the new skeletons are in world-coordinates, cross-setup generalization is expected to be relatively straightforward for the models. The *NTU-RGBD-120* dataset also has a *xset* split that was intended to test this. But it has the problem that some new setups also contain new subjects as well, so it does not specifically test the cross-setup generalization alone. This makes it difficult to isolate only the effect of setup changes, so the accuracy results provide no directly usable insights, and for those reasons no evaluation was done on the *xset* split. Besides that, the *xsub* split already includes setup switches as well.

Instead of just using differences in the inputs, another option is to also use differences in the model architectures, by ensembling the different models as well. Table X shows that this can result in further improvements, but as before, at the cost of real-time performance.

Method	60	120
MSG3D + DG-STGCN + ProtoGCN	97.9	96.0

TABLE X: Ensembling different model architectures.

D. About objects and image inputs

While this work does not explore it, the benefit of multi-view inputs should also be applicable to approaches that use color images for action recognition as well. Especially with the *NTU-RGBD* dataset, the additional images contain valuable information about objects that are being manipulated in some actions, which is not available in the skeleton data. Distinguishing whether a person is drinking a cup of water or eating a hamburger is quite difficult with skeleton data only, but should be much easier when the image data is available as well. The confusion matrices also show a higher error rate at many actions which involve objects.

A possible downside of such a combined model could be that, because the recognition model will get more input data, it will lead to lower few-shot performance, because the model focuses more on the object appearance than the motion patterns. But this is just speculation at this point, and should be investigated in future work. A further downside would be the increased computational cost, which is an important factor in real-time applications. For example in the robotic application of *Tutabo* [4], the pose estimations are calculated on edge devices, because sending the image streams to a central computer would already be too time-consuming. So a combined image+skeleton model would also need to run on the edge device, which could notably increase its computational requirements.

A possible option in such a scenario, and one that would also fit the skeleton-based input concept, could be to use object bounding-box detections, triangulate them similarly to the skeletons, and use them as additional input to the action recognition model. This also would be a privacy-friendly approach for distributed or non-local applications, because only the sparse skeleton and object position data would need to be transmitted, instead of the full images. But again, this will be left for future work.

E. Confusion Matrices

The confusion matrices in Figure 3 and Figure 4 show that most actions are classified very well, and confusions often occur between similar ones. In many cases the (here not existing) knowledge about objects could help the model, for example to distinguish between *eat snack* and *drink water* or between *reading* and *play with tablet*.

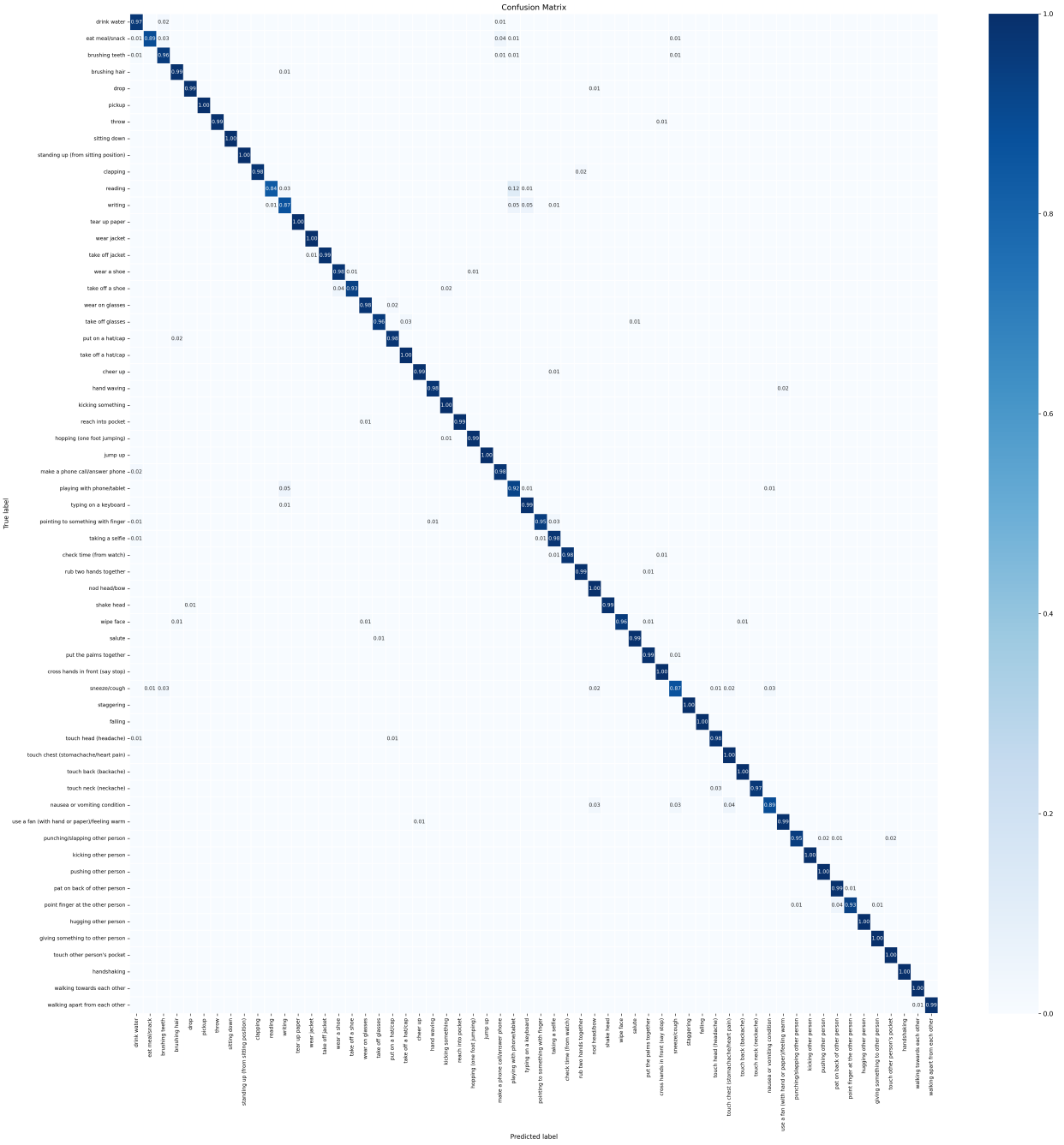

Fig. 3: Confusion matrix of the ensembled *ProtoGCN* model on *NTU-RGBD-60-xsub*.

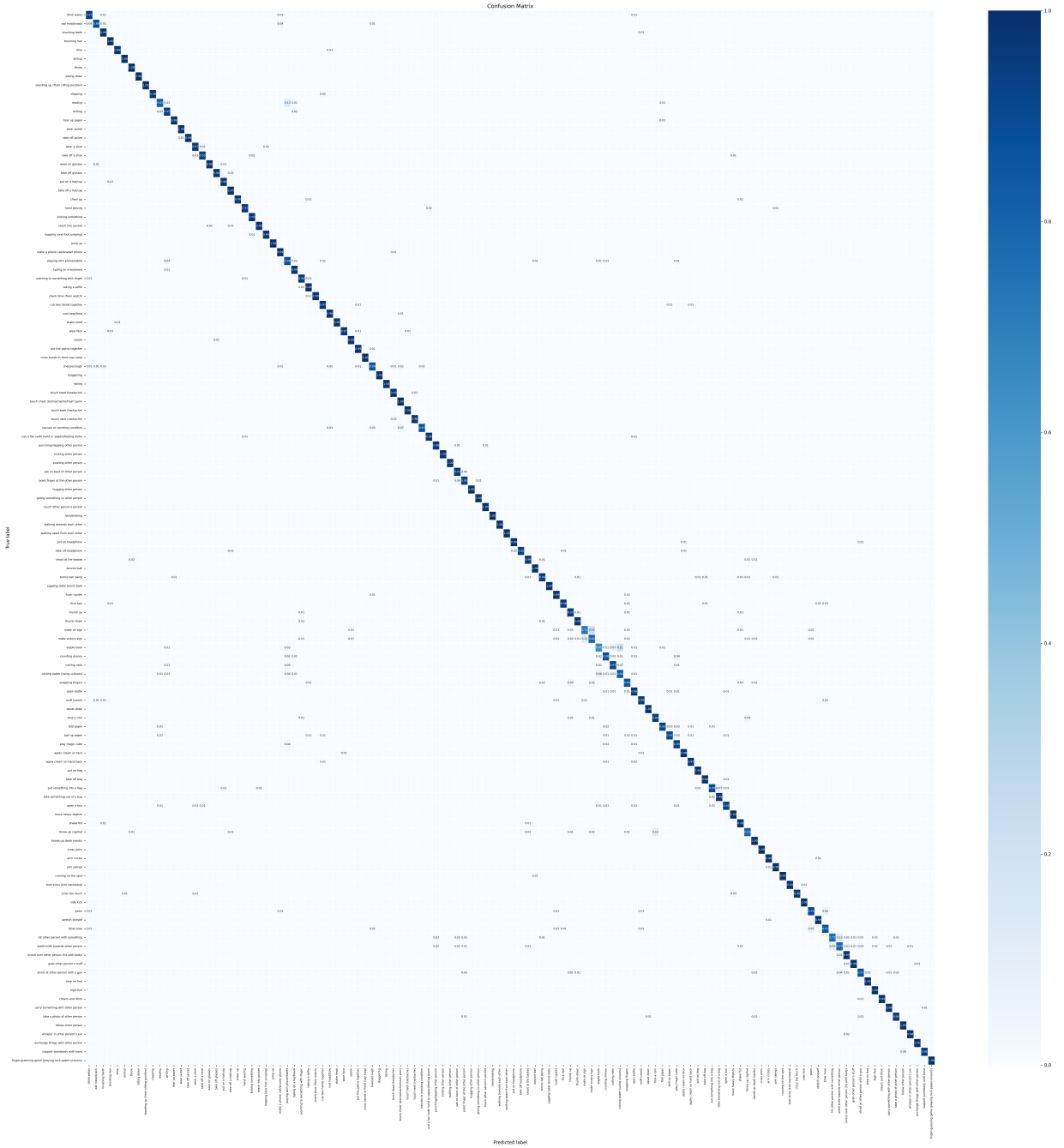

Fig. 4: Confusion matrix of the ensembled *ProtoGCN* model on *NTU-RGBD-120-xsub*.