A Transformer-Based Pipeline for Efficient Hand Gesture Recognition

Athanasios Papanikolaou[1], Vladimir Sivtsov[1], Daniil Shkolnik[1], Enrica Zereik[2], Ivan Markovic[1], Ivan Petrovic[1], and Fabio Bonsignorio[1]

University of Zagreb Faculty of Electrical Engineering and Computing,
email: name.surname@fer.unizg.hr,
Italian National Research Council, Institute of Marine Engineering (CNR-INM),
16149, Genoa, Italy,
email: enrica.zereik@cnr.it

Abstract. This work presents a hierarchical multi-model hand gesture recognition system for structured motion segmentation in robotic manipulation and disassembly. Unlike single-stage classification, this system progressively refines motion understanding across multiple levels. It offers a scalable and interpretable solution for structured gesture recognition, applicable to robotic automation, human-robot collaboration, and assistive technologies. The pipeline integrates TimeSformer, Mediapipe hand tracking, Kalman filtering, and Smart Frame Selection to enhance accuracy and computational efficiency. Gestures are classified in stages, first identifying broad gesture types before refining grasp power, hand contact type, and post-release object behaviour. This structured approach improves interpretability and mitigates class imbalance issues in traditional multi-class models. A dataset tailored for hierarchical classification ensures the system generalizes beyond isolated training samples. Experimental results demonstrate strong classification performance across levels, with a modular design enabling seamless expansion.

Keywords: Gesture Recognition · TimeSformer · Transformer Networks · Human-Robot Interaction · Deep Learning.

1 Introduction

Recent advancements in AI, machine learning, and computer vision have significantly transitioned industrial robotics from rigid automation to more adaptable systems capable of handling unpredictable environments [1]. Among these advancements, robotic disassembly has become a key process in sustainable manufacturing. However, its implementation remains challenging due to variations in object conditions, placement, and fastening methods, making full automation difficult [2], [3]. Traditional planning approaches require extensive programming but often struggle to generalize across diverse products. An alternative solution is learning from human demonstrations, which enables robots to adapt their behaviour dynamically [4]. However, translating human actions into

robot-executable sequences requires structured motion segmentation, ensuring gestures are decomposed into discrete, analyzable components. To address this challenge, researchers have long relied on Therbligs, a taxonomy of fundamental motion elements used to optimize manual labour [5]. Modern frameworks build upon this concept by categorizing actions based on force, hand contact, and post-release object behaviour [6], [7]. However, these approaches traditionally depend on manual annotation, which limits scalability. Deep learning-based gesture recognition provides an alternative by automating segmentation while preserving structured classification [8]. Transformer-based architectures such as TimeSformer [9] have demonstrated improvements in spatiotemporal modelling, surpassing convolutional methods in gesture recognition. Hierarchical classification further refines segmentation by breaking down actions into progressively finer details, as seen in FineGym [10] and Motion Programs [11].

Expanding on prior work [12], this paper introduces a hierarchical multi-model pipeline integrating TimeSformer, Smart Frame Selection [13], Mediapipe tracking [14], and Kalman filtering [15]. Unlike single-stage models, our proposed system progressively classifies gestures, first identifying broad motion types before refining aspects such as grasp intensity, contact type, and post-release dynamics. By structuring gesture classification hierarchically, this approach enhances robotic disassembly through a more interpretable motion representation. Though primarily focused on reverse manufacturing, this method is also adaptable to assembly, teleoperation, and assistive robotics, improving gesture-based robotic learning in dynamic environments.

2 Related Work

Advancements in deep learning have significantly improved gesture recognition, striking a balance between accuracy and efficiency [16]. Robotic gesture classification has evolved beyond flat recognition models, shifting towards hierarchical modelling where actions are structured into dependent sub-actions [17]. Multi-stage temporal convolutional networks (MS-TCN) [18] and non-local neural networks [19] enhance motion understanding by modelling sequential relationships. However, real-world gesture segmentation presents significant challenges due to motion variability, environmental shifts, and differences in user behaviour [20]. Hierarchical motion structuring mitigates these challenges, as demonstrated in FineGym [10], where high-level activities are broken down into multi-level sub-actions. Similarly, Smart Frame Selection [13] enhances computational efficiency by identifying keyframes instead of processing full video sequences.

Beyond deep learning, motion primitives improve robotic task planning by decomposing actions into fundamental units. Therbligs-based segmentation has been used in robotic learning to enhance gesture classification interpretability [5]. Recent hierarchical taxonomies have further refined this segmentation process by classifying motion based on grasp force, contact area, and post-release behaviour, creating a scalable framework for robotic automation [6]. Hierarchical segmentation is particularly relevant for human-robot collaboration, where

robots must adapt their behaviour based on human demonstrations rather than relying on rigid programming [21], [2]. The Functional and Practical Taxonomy for Reverse Manufacturing [6] exemplifies how multi-tiered classification can align gesture recognition with robotic task planning.

Traditional manual annotation is impractical for large-scale motion segmentation. Temporal convolutional networks (TCNs) [18] and self-attention models [9] automate segmentation but are typically constrained by flat classification structures. Our work aims to improve gesture classification precision by integrating deep learning with hierarchical segmentation while enhancing robotic adaptability.

3 Method

This work presents a hierarchical gesture recognition framework for robotic disassembly and fine-grained manipulation analysis. Unlike single-stage classification, this system progressively refines motion understanding by distinguishing attributes such as grasp force, hand contact type, and post-release object behaviour. The dataset follows structured segmentation frameworks, including Therbligs [5] and the Functional and Practical Taxonomy for Reverse Manufacturing [6], ensuring compatibility with robotic task execution.

3.1 Dataset

The dataset was specifically designed and custom-recorded for this study to support hierarchical classification by progressively refining gesture attributes rather than assigning a single label per motion. Collected in controlled indoor environments with varying lighting, backgrounds, and object interactions, it ensures adaptability to diverse conditions. Gestures were performed naturally, without predefined constraints, to better simulate real-world robotic manipulation tasks.

The dataset consists of multiple subsets corresponding to different classification stages. At the first level, gestures are broadly categorized into Grasp, Release, Hold, Move, Reach, Orient, Press, and Idle. The second level refines grasping motions by determining grasp power—Powerful, Intermediate, or Precision—while for release actions, it predicts post-release behaviour, categorizing movements as Stationary, Passive Movement, or Active Push. The third level further distinguishes grasping gestures based on hand contact type, identifying whether the contact occurs with the Palm, Pad, or Side. This modular approach simplifies training by distributing tasks across specialized models, preventing excessive dataset expansion while maintaining classification granularity. Unlike existing datasets, this homemade collection was explicitly structured for hierarchical classification, allowing precise motion segmentation across refinement levels. Future extensions could incorporate additional segmentation levels, such as detailed finger positioning, to further enhance gesture classification precision [6].

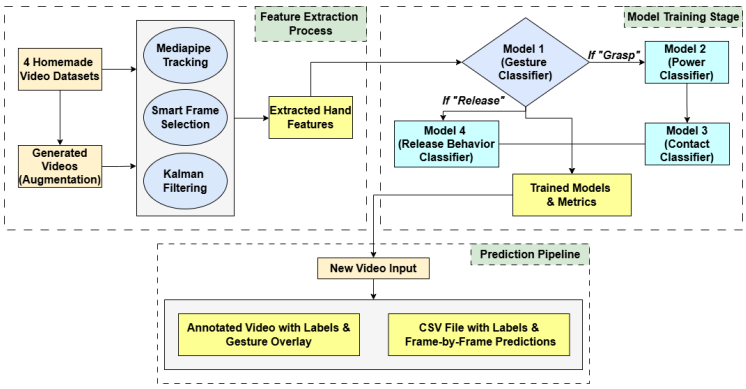


Fig. 1. Overview of the proposed hierarchical hand gesture recognition pipeline.

To ensure robust generalization, recordings were collected in diverse environments with controlled variations in background, lighting, and camera angles with the help of a second observer. Everyday objects were included for contextual relevance. Augmentation techniques such as flipping, rotation, distortions, and brightness adjustments expanded dataset diversity tenfold while preserving gesture integrity. The final dataset consists of 263 minutes of recordings and 36,720 frames, balancing control and scalability. Unlike large-scale datasets like EPIC-KITCHENS-100 [22], which focus on unstructured activities over 100 hours of video and 20 million frames, this dataset is optimized for structured, fine-grained gesture recognition, making it well-suited for robotic learning, industrial automation, and human-robot collaboration [21]. Figure 1 provides an overview of the recognition pipeline, covering preprocessing, model training, hierarchical classification, and structured output generation.

3.2 Preprocessing Pipeline

The preprocessing pipeline transforms raw video data into structured inputs, ensuring high-quality representations for hierarchical gesture classification. This process consists of four key stages. First, Smart Frame Selection [13] extracts keyframes that capture critical motion transitions, reducing redundancy and computational load while preserving gesture integrity. This selection mechanism ensures that only the most relevant frames are processed, minimizing unnecessary computations while maintaining essential gesture dynamics. Next, hand tracking and feature extraction are performed using Mediapipe Hands [14], which detects 21 landmarks through an RGB-based pipeline. This enables efficient real-time execution without requiring depth sensors. The two-stage process involves palm detection followed by landmark estimation, allowing for continuous hand motion tracking essential for structured classification.

To further refine tracking accuracy, motion smoothing with Kalman Filtering [15] is applied to reduce noise and stabilize motion trajectories. Additionally, the Extended Kalman Filter (EKF) [23] is used to refine depth variations, enhancing robustness in cases where occlusions or sudden hand movements could

introduce instability. Finally, hierarchical processing ensures that preprocessed data is passed through a multi-stage classifier, progressively refining gesture attributes at different classification levels, as illustrated in Figure 1.

This pipeline enhances recognition accuracy by eliminating noise and improving motion representation. Its structured approach ensures adaptability across different environments, making it particularly well-suited for robotic disassembly, teleoperation, and industrial automation, where precise gesture classification is crucial for effective human-robot interaction.

3.3 Model Architecture and Training

This system utilizes TimeSformer, a transformer-based architecture optimized for spatiotemporal gesture classification in video sequences. TimeSformer processes video data using self-attention mechanisms across spatial and temporal dimensions, unlike CNNs, which depend on local receptive fields and spatiotemporal convolutions to extract motion features. This architecture eliminates the cubic complexity associated with 3D CNNs, making it computationally efficient while preserving strong motion modelling capabilities. Given an input video $X \in R^{T\times H\times W\times C}$, where T represents the number of frames, and H, W, C represent the height, width, and number of channels, TimeSformer tokenizes the input into non-overlapping patches and applies self-attention independently in spatial and temporal dimensions.

TimeSformer employs a Divided Attention mechanism, where spatial attention is applied within each frame, followed by temporal attention across frames. This factorization reduces computational complexity from:

$$O((THW)^2) \quad \text{to} \quad O(T^2) + O(HW)^2, \tag{1}$$

significantly improving scalability without sacrificing accuracy [9]. The spatial feature representation of each patch is computed using a linear projection, resulting in a sequence of patch embeddings $z \in R^{N\times D}$, where $N = \frac{HW}{p^2}$ represents the total number of patches, and D is the embedding dimension. The Multi-Head Self-Attention (MHSA) mechanism computes attention scores using:

$$\text{MHSA}(Q, K, V) = \text{Concat}(H_1, H_2, ..., H_h)W^O, \tag{2}$$

where each attention head computes $H_i = \text{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i$, and $d_k = \frac{D}{h}$ represents the feature dimension per head. The output of the attention mechanism is passed through layer normalization and residual connections, forming the core of the transformer block.

Compared to ViViT [24], which also factorizes spatial-temporal attention, TimeSformer improves efficiency by explicitly separating spatial and temporal processing. Unlike I3D [25], which models motion using 3D convolutions, TimeSformer avoids parameter explosion, making it more suitable for real-time gesture recognition. To leverage these benefits, the system trains four specialized TimeSformer models, each focusing on a distinct aspect of motion classification. The

first model categorizes gestures into broad types such as Grasp, Release, Move, and Hold. The second model refines grasping motions by distinguishing power levels: Powerful, Intermediate, or Precision. The third model classifies contact types, identifying whether the grasp occurs using the Palm, Pad, or Side of the hand. The fourth model predicts post-release object behaviour, determining whether the object remains Stationary, Moves Passively, or Moves Actively. Each model is trained independently using curated subsets to prevent class imbalance. During inference, frames are sequentially processed through these models, progressively refining classification details.

Several training optimizations were implemented to enhance computational efficiency. Gradient Checkpointing reduces memory usage by recomputing intermediate activations, allowing for larger batch sizes. Automatic Mixed Precision (AMP) utilizes half-precision floating points, accelerating training while maintaining numerical stability. Dropout Regularization is applied within transformer layers to prevent overfitting, and Preprocessed Tensor Storage eliminates redundant computations by storing processed videos as tensors before training.

The training utilized the Adam optimizer with an initial learning rate of $1e^{-4}$, progressively reduced via a cosine decay schedule. Cross-entropy loss was employed for multi-class classification, and hyperparameters were fine-tuned via grid search for batch size, learning rate, and attention configurations.

4 Results

The proposed hierarchical gesture recognition system was evaluated on classification accuracy, generalization ability, and computational efficiency. This section presents the training and validation performance of the four models, classification performance on unseen test data, and an analysis of structured multi-model predictions.

4.1 Training and Validation Performance

A stratified dataset split across all datasets(70% training, 15% validation, 15% testing) ensured unbiased performance evaluation. Unlike single-stage classification, this system employs four independent models, each dedicated to a specific classification level. This modular training prevents cross-task interference, efficiently ensuring that each model learns distinct motion attributes while maintaining overall system coherence.

All models were trained for 50 epochs with a batch size of 8 and an initial learning rate of 0.0001, progressively reduced using a cosine decay schedule. Early stopping was applied to prevent overfitting once validation accuracy plateaued. The models demonstrated stable convergence, with training and validation accuracy exceeding 94% and 90%, respectively, across all levels. Training loss values remained always below 0.01, confirming effective optimization, while validation losses stabilized within an acceptable range below 0.2. Overall, results indicated strong classification performance with minimal overfitting.

Loss curves showed a steady decline in training loss and stabilization in validation loss, reinforcing the effectiveness of the implemented training strategies. Overfitting was mitigated through dropout regularization, gradient checkpointing, and AMP, which collectively optimized GPU utilization and improved generalization.

To enhance computational efficiency, preprocessing and tensor caching minimized redundant computations by storing preprocessed frames, significantly reducing memory overhead. Gradient checkpointing further lowered memory consumption by recomputing intermediate activations, allowing for larger batch sizes. Meanwhile, AMP leveraged half-precision floating points to accelerate computation while maintaining numerical stability, and dropout regularization was applied within transformer layers to prevent overfitting.

These optimizations reduced training time, allowing each epoch to be completed in under three minutes using one GPU NVIDIA RTX™ 6000 Ada Generation of 64GB RAM . The modular design enhances scalability, ensuring computational feasibility as additional classification levels are introduced. Furthermore, these improvements facilitate real-time deployment by minimizing the computational cost of processing video sequences while maintaining high classification accuracy.

4.2 Classification Performance

The predictive performance of each classification model was evaluated using precision, recall, and F1-score. Unlike single-stage gesture recognition, our system progressively refines predictions across multiple levels. Evaluations were repeated 100 times with different randomly sampled test sets to ensure robustness. Table 1 reports the average metrics, offering a reliable estimate of generalization.

All models achieved macro-average F1-scores above 0.97, indicating strong and consistent performance. By decomposing gesture recognition into specialized sub-tasks, the hierarchical structure mitigates class imbalance and improves model interpretability. This progressive classification approach also supports error isolation across levels, making it especially suited for robotics applications requiring semantic granularity.

4.3 Comparison to Existing Approaches

We compare the proposed hierarchical system to contextualise performance against existing gesture and action recognition models. Table 2 presents a comparative analysis, highlighting accuracy and efficiency advantages.

In contrast to TimeSformer [9] and ViViT [24], which require extensive datasets and computational resources, our system achieves comparable accuracy (98.3%) with significantly greater efficiency. Its multi-model structure enables progressive, multi-level classification rather than relying on a single-step prediction. This aligns with structured motion segmentation and enhances interpretability, making it effective for robotics tasks requiring fine-grained gesture

Gesture Model			
Gesture	**Precision**	**Recall**	**F1-Score**
Grasp	0.883	1.000	0.935
Hold	0.982	1.000	0.991
Move	1.000	0.986	0.993
Release	0.977	1.000	0.988
Orient	1.000	0.978	0.989
Press	1.000	0.987	0.993
Reach	1.000	0.896	0.943
Idle	1.000	0.980	0.990
Macro Avg	0.993	0.991	0.992
Weighted Avg	0.994	0.993	0.993

Power Model			
Power Level	**Precision**	**Recall**	**F1-Score**
Powerful	0.981	0.990	0.985
Intermediate	0.976	0.937	0.954
Precision	0.960	0.988	0.973
Macro Avg	0.972	0.972	0.972
Weighted Avg	0.975	0.975	0.975

Hand Contact Model			
Contact Type	**Precision**	**Recall**	**F1-Score**
Palm	0.966	0.988	0.976
Pad	0.991	1.000	0.995
Side	0.990	0.958	0.973
Macro Avg	0.982	0.982	0.982
Weighted Avg	0.985	0.985	0.985

Release Model			
Release Type	**Precision**	**Recall**	**F1-Score**
Stationary	0.992	1.000	0.996
Passive Move	1.000	0.954	0.976
Active Push	0.963	1.000	0.981
Macro Avg	0.985	0.985	0.985
Weighted Avg	0.987	0.987	0.987

Table 1. Classification performance metrics (precision, recall, F1-score) for each model averaged over 100 independent runs with different randomly sampled test datasets.

understanding. By distributing complexity across stages, the system also mitigates class imbalance—an issue common in large flat classifiers.

Compared to MS-TCN++ [18], which uses temporal convolutions, our transformer-based models capture long-range dependencies without needing long sequential data. While TCNs are efficient, their fixed receptive fields limit adaptability in variable or overlapping gestures. Transformers, by contrast, dynamically model global relationships, improving recognition in unstructured environments. This flexibility makes our approach more robust for real-world deployment.

Unlike motion primitive-based methods like Therbligs in Action [27], which depend on hand-crafted decomposition, our system integrates such structured analysis into a deep learning pipeline, combining interpretability with adaptability. Similarly, large-scale datasets like EPIC-KITCHENS-100 [22] are designed for unconstrained, long-term activity understanding and lack the hierarchical labels needed for progressive segmentation. Our custom dataset, although smaller, is tailored to structured gesture refinement, supporting high-confidence evaluation across classification levels.

Model	**Accuracy**	**F1-Score**	**Dataset**	**Speed/Epoch**
Proposed Model	98.3%	98.2%	Homemade, Augmented	≤3m
TimeSformer [9]	62.1%	55.4%	Large (Kinetics, EPIC)	Slow
ViViT [24]	62.4%	56.0%	Large-Scale (Kinetics, EPIC)	Multi-Stage
MS-TCN++ [18]	88.8% (GTEA)	85.7%	Moderate (50 Salads, GTEA)	≤2m
Tsinganos et al. [26]	Lower	N/A	Small (EMG)	Real-Time
Therbligs in Action [27]	64.03%	61.46%	Moderate (EPIC)	Moderate
EPIC-KITCHENS-100 [22]	N/A	N/A	Large-Scale (100h)	N/A

Table 2. Comparison of the proposed model with existing gesture/action recognition approaches.

While Table 2 includes widely used models for reference, these results are not intended as direct benchmarks. Flat, monolithic architectures trained on large-scale datasets differ fundamentally from our hierarchical system. Our objective is not to compete on raw dataset size, but to show that reliable, multi-level classification can be achieved using a compact, well-designed dataset purpose-built for structured recognition. This makes our pipeline especially relevant for robotics contexts where interpretability, control, and semantic segmentation are essential.

4.4 Final Prediction Analysis

The trained models were tested on an extended video containing continuous hand gestures, novel objects, and varying background conditions to assess real-world generalization. Unlike the training data, which featured isolated gestures, this sequence included natural, spontaneous transitions and dynamic hand trajectories, introducing a more realistic testing scenario. Despite these challenges—such as novel objects and unseen lighting—the system maintained high recognition accuracy across most frames, demonstrating robust generalization to unconstrained settings.

The hierarchical pipeline processed each frame sequentially: the Gesture Type Model first categorized the motion, followed by refinements from the Power, Hand Contact, and Release Behaviour models. Frame-by-frame predictions were recorded in a structured CSV file, with temporal filtering applied to smooth transient errors. Frames without detected hands were labelled "N/A" to prevent false positives.

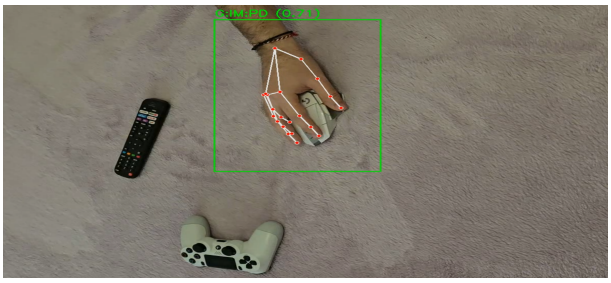


Fig. 2. Example Frame of Multi-Level Gesture Prediction.

An annotated video visualized predictions in real time, using structured gesture labels and confidence scores, as illustrated in Figure 2. Each hand was enclosed in a bounding box, with key landmarks displayed to convey pose and articulation. This multi-level classification enabled fine-grained gesture interpretation by progressively refining motion attributes.

While the system successfully recognized most gestures, it did not fully replicate the near-perfect accuracy observed in controlled training. Rapid transitions occasionally caused misclassifications—particularly between Intermediate and Precision grasps, or in distinguishing passive from active object release. One

current limitation is that error propagation across classification levels has not been formally analyzed. Although downstream predictions remained mostly coherent during qualitative evaluation, future work should quantify the impact of early-stage misclassifications on deeper layers.

Further improvements could explore temporal attention or adaptive smoothing to improve stability during fast transitions. Extending the dataset with more dynamic gestures and object types would also help enhance robustness in complex environments.

5 Conclusion

In this paper we have presented a hierarchical multi-model framework for hand gesture recognition, integrating TimeSformer, Mediapipe tracking, Kalman filtering, and Smart Frame Selection to enhance structured motion analysis. Unlike single-stage classification approaches, this system progressively refines motion interpretation through a hierarchical processing pipeline.

Evaluations on a structured dataset designed for hierarchical classification confirmed the system's effectiveness across multiple levels. The hierarchical structure improves classification accuracy while mitigating class imbalances and computational inefficiencies. The system successfully processed continuous, real-world gesture sequences, demonstrating its ability to generalize beyond isolated training samples. However, challenges remain in distinguishing subtle hand posture variations and adapting to rapid gesture transitions, particularly in precision grasping and post-release dynamics.

Future research will focus on enhancing transition modelling through temporal attention strategies and adaptive filtering to reduce errors in sequential movements. Expanding the dataset to include additional grasping conditions and environmental variations will further improve generalization. The system's modular nature allows seamless integration of new classification levels, such as finger articulation, enabling more complex manipulation tasks. The proposed hierarchical framework also lends itself to practical deployment in robotic systems. For instance, in a disassembly cell, the ability to detect not just a grasp but also its intensity and contact type can inform force control strategies and tool selection. In assistive robotics, the system could track subtle grip changes during rehabilitation, offering therapists quantifiable insights into patient recovery progress. These concrete use cases underline the system's potential for real-world impact in both industrial and healthcare domains. Incorporating imitation learning techniques could further refine structured action learning from human demonstrations.

Beyond robotic disassembly, the proposed framework applies to industrial automation, human-robot collaboration, and assistive robotics. More broadly, this research advances structured motion understanding, providing insights relevant to augmented reality, healthcare and intelligent user interfaces, bridging the gap between low-level motion detection and high-level task execution.

6 Acknowledgments

This work was supported by the H2020 project AIFORS under Grant Agreement No 952275.

References

1. Andrius Dzedzickis, Jurga Subaciute-Zemaitiene, Ernestas Sutinys, Urte Samukaite-Bubniene, and Vytautas Bucinskas. Advanced applications of industrial robotics: New trends and possibilities. *Applied Sciences*, 12(1):135, dec 2021.
2. Jun Huang, Duc Truong Pham, Yongjing Wang, Mo Qu, Chunqian Ji, Shizhong Su, Wenjun Xu, Quan Liu, and Zude Zhou. A case study in human-robot collaboration in the disassembly of press-fitted components. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 234(3):654–664, oct 2019.
3. Paul Vanegas, Jef R. Peeters, Dirk Cattrysse, Paolo Tecchio, Fulvio Ardente, Fabrice Mathieux, Wim Dewulf, and Joost R. Duflou. Ease of disassembly of products to support circular economy strategies. *Resources, Conservation and Recycling*, 135:323–334, 2018.
4. Yajing Zang, Pengfei Wang, Fusheng Zha, Wei Guo, Chao Zheng, and Lining Sun. Peg-in-hole assembly skill imitation learning method based on promps under task geometric representation. *Frontiers in Neurorobotics*, 17:1320251, 2023.
5. Eadom Dessalene, Michael Maynord, Cornelia Fermuller, and Yiannis Aloimonos. Therbligs in action: Video understanding through motion primitives. 2023.
6. Annagiulia Morachioli, Vladimir Sivtsov, Nicolas Rojas, and Fabio Bonsignorio. A functional and practical taxonomy for the industrial implementation of highly automated reverse manufacturing cells. *IEEE Open Journal of the Industrial Electronics Society*, 5:1115–1139, 2024.
7. Vladimir Sivtsov, Athanasios Papanikolaou, Annagiulia Morachioli, Ivan Markovic, Ivan Petrovic, Enrica Zereik, and Fabio Bonsignorio. Action primitives for bimanual robotic disassembly cell of printer cartridges. Submitted, 2025.
8. Zhaoyong Hu, Shuquan Sun, Yueming Wu, Hansheng Yan, and Teng Zhu. Research on gesture recognition and interaction of virtual collaborative disassembly training. In *2021 IEEE 7th International Conference on Virtual Reality (ICVR)*, pages 246–253. IEEE, 2021.
9. Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 813–824. PMLR, 2021.
10. Dian Shao, Yue Zhao, Bo Dai, and Dahua Lin. Finegym: A hierarchical video dataset for fine-grained action understanding. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2613–2622, 2020.
11. Sumith Kulal, Jiayuan Mao, Alex Aiken, and Jiajun Wu. Hierarchical motion understanding via motion programs. *Computer Vision and Pattern Recognition*, 2021.
12. Athanasios Papanikolaou, Vladimir Sivtsov, Annagiulia Morachioli, Ivan Markovic, Ivan Petrovic, Enrica Zereik, and Fabio Bonsignorio. Design of a bimanual disassembly cell for the cartridge substitution of an off-the-shelf office printer. Submitted, 2025.

13. Shreyank N Gowda, Marcus Rohrbach, and Laura Sevilla-Lara. Smart frame selection for action recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(2):1451–1459, may 2021.
14. Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. Mediapipe hands: On-device real-time hand tracking, 06 2020.
15. Qiushi Fu and Marco Santello. Tracking whole hand kinematics using extended kalman filter. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pages 4606–4609, 2010.
16. Dan Kondratyuk, Liangzhe Yuan, Yandong Li, Li Zhang, Mingxing Tan, Matthew Brown, and Boqing Gong. Movinets: Mobile video networks for efficient video recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16015–16025, 2021.
17. Quan Liu, Zhihao Liu, Wenjun Xu, Quan Tang, Zude Zhou, and Duc Truong Pham. Human-robot collaboration in disassembly for sustainable manufacturing. *International Journal of Production Research*, 57(12):4027–4044, 2019.
18. Shijie Li, Yazan Abu Farha, Yun Liu, Ming-Ming Cheng, and Juergen Gall. Ms-tcn++: Multi-stage temporal convolutional network for action segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):6647–6658, jun 2023.
19. Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017.
20. Kai Hu, Chaowen Shen, Tianyan Wang, Keer Xu, Qingfeng Xia, Min Xia, and Chengxue Cai. Overview of temporal action detection based on deep learning. *Artificial Intelligence Review*, 57(2), feb 2024.
21. Dagmar Reinhardt and Lynn Masuda. Therblig to robot: Action packages, robot motion and human-robot collaboration in domestic environments. *SPOOL*, 11(1):49–64, 2024.
22. Dima Aldamen, Davide Moltisanti, Evangelos Kazakos, Hazel Doughty, Jonathan Munro, William Price, Michael Wray, Tobias Perrett, and Jian Ma. Epic-kitchens-100, 2020.
23. Sangheon Park, Sunjin Yu, Joongrock Kim, Sungjin Kim, and Sangyoun Lee. 3d hand tracking using kalman filter in depth space. *EURASIP Journal on Advances in Signal Processing*, 2012(1), feb 2012.
24. Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6816–6826, 2021.
25. João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4733, 2017.
26. Panagiotis Tsinganos, Bart Jansen, Jan Cornelis, and Athanassios Skodras. Real-time analysis of hand gesture recognition with temporal convolutional networks. *Sensors*, 22(5):1694, feb 2022.
27. Eadom Dessalene, Michael Maynord, Cornelia Fermuller, and Yiannis Aloimonos. Therbligs in Action: Video Understanding through Motion Primitives . In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10618–10626, Los Alamitos, CA, USA, Jun 2023. IEEE Computer Society.