

On the Recognition of Human Activities and the Evaluation of Its Imitation by Robotic Systems

Approved Dissertation thesis for the partial fulfilment of
the requirements for a
Doctor of Natural Sciences (Dr. rer. nat.)
Fachbereich 4: Informatik
Universität Koblenz

by

M. Sc. Raphael Memmesheimer

Chair of PhD Board: Prof. Dr. Ralf Lämmel
Chair of PhD Commission: Prof. Dr. Matthias Gouthier
Examiner and Supervisor: Prof. Dr. Dietrich Paulus
Further Examiners: Prof. Dr. Hildegard Kühne

Date of the doctoral viva: 28.11.2022

Acknowledgements

First of all, I want to thank Prof. Dr. Dietrich Paulus for a great environment to develop as an independent researcher. I had a great time in the nerdy (in a positive sense) and fun group you shaped. Further, I would also like to thank Prof. Dr. Hilde Kühne, as she kindly agreed to serve as an external reviewer. Special thanks go to Nick Theisen for fruitful discussions, almost always when I stepped into your door with a coarsely crafted idea that I had a hard time to even paraphrasing and doodling, I left the door with a bag-of-ideas to develop. In a sense, this small area between the door and the always noisy whiteboard was a real innovation area for me. Also, a big thanks to Viktor Seib for all the discussions, the proofreading, the early mentoring from my bachelor's to my master's and for handing me the Robbie project in such a great condition. I also want to thank Ivanna Kramer for her support, vital discussions, and the fantastic research stay in Lisbon. During my PhD. studies, I felt surrounded by many great students that took part in the project- and research internships or wrote bachelor- and master theses under my supervision. You realized many of the ideas much better than I ever could. I want to express my special gratitude to Michael Duhme and Simon Häring for their great support by extending their theses to publications. Thanks to my friends (Hannah, Helena, Leoni, Lina, Lisa, Marike, Max, Sandra, Stephan, Tim, Vanessa, Veronika and Wilko) for the balance throughout the last years and the fun we had through countless events, social nights, trips and even talking. I want to express my profound gratitude to my family for their support over the years – and, of course, Jasmin, thanks for joining the ride and your unlimited support in so many aspects of life. Thank you!

Abstract

This thesis addresses the problem of action recognition through the analysis of human motion and the benchmarking of its imitation by robotic systems. For our action recognition related approaches, we focus on presenting approaches that generalize well across different sensor modalities. We transform multivariate signal streams from various sensors to a common image representation. The action recognition problem on sequential multivariate signal streams can then be reduced to an image classification task for which we utilize recent advances in machine learning. We demonstrate the broad applicability of our approaches formulated as a supervised classification task for action recognition, a semi-supervised classification task for one-shot action recognition, modality fusion and temporal action segmentation.

For action classification, we use an EfficientNet Convolutional Neural Network (CNN) model to classify the image representations of various data modalities. Further, we present approaches for filtering and the fusion of various modalities on a representation level. We extend the approach to be applicable for semi-supervised classification and train a metric-learning model that encodes action similarity. During training, the encoder optimizes the distances in embedding space for self-, positive- and negative-pair similarities. The resulting encoder allows estimating action similarity by calculating distances in embedding space. At training time, no action classes from the test set are used.

Graph Convolutional Network (GCN) generalized the concept of CNNs to non-Euclidean data structures and showed great success for action recognition directly operating on spatio-temporal sequences like skeleton sequences. GCNs have recently shown state-of-the-art performance for skeleton-based action recognition but are currently widely neglected as the foundation for the fusion of various sensor modalities. We propose incorporating additional modalities, like inertial measurements or RGB features, into a skeleton-graph, by proposing fusion on two different dimensionality levels. On a channel dimension, modalities are fused by introducing additional node attributes. On a spatial dimension, additional nodes are incorporated into the skeleton-graph.

Transformer models showed excellent performance in the analysis of sequential data. We formulate the temporal action segmentation task as an object detection task and use a detection transformer model on our proposed motion image representations. Experiments for our action recognition related approaches are executed on large-scale publicly available datasets. Our approaches for action recognition for various modalities, action recognition by fusion of various modalities, and one-shot action recognition demonstrate state-of-the-art results on some datasets.

Finally, we present a hybrid imitation learning benchmark. The benchmark consists of a dataset, metrics, and a simulator integration. The dataset contains RGB-D image sequences of humans performing movements and executing manipulation tasks, as well as the corresponding ground truth. The RGB-D camera is calibrated against a motion-capturing system, and the resulting sequences serve as input for imitation learning approaches. The resulting policy is then executed in the simulated environment on different robots. We propose two metrics to assess the quality of the imitation. The trajectory metric gives insights into how close the execution was to the demonstration. The effect metric describes how close the final state was reached according to the demonstration. The Simitate benchmark can improve the comparability of imitation learning approaches.

Kurzfassung

In dieser Arbeit präsentieren wir Ansätze zur Aktionserkennung durch die Analyse menschlicher Bewegung sowie dem Benchmarking von Imitation beobachteter Aktionen durch Roboter. Unsere Aktionserkennungsansätze legen einen Fokus auf die Generalisierung über verschiedene Modalitäten. Wir transformieren multivariate Signalsequenzen von verschiedenen Sensoren in eine einheitliche Bildrepräsentation. Dadurch kann das Aktionserkennungsproblem verschiedener Modalitäten zu einem Bildklassifikationsproblem reduziert werden. Für die Klassifikation der Repräsentationen bauen wir auf den Fortschritten des maschinellen Lernens auf. Wir zeigen eine breite Anwendbarkeit unserer Ansätze, formuliert als Probleme des überwachten Lernens und des teilüberwachten Lernens zum Klassifizieren der Aktionen, Klassifizieren der Aktionen mittels weniger Referenzbeispiele, Sensordatenfusionierung und temporaler Aktionssegmentierung.

Zur Klassifikation der Aktionen, auf Basis unserer einheitlichen Repräsentation für verschiedene Modalitäten, nutzen wir ein Modell basierend auf einem EfficientNet Faltungsnetz. Weiterhin stellen wir Ansätze zum Filtern und Fusionieren verschiedener Modalitäten auf einer Repräsentationsebene vor. Dieser überwachte Lernansatz für die Aktionserkennung wird anschließend erweitert zu einem Ansatz des teilüberwachten Lernens. Dazu verwenden wir einen Ansatz zum Metriklernen. Dieser transformiert die Repräsentationen in einen Einbettungsraum, in welchem Aktionsähnlichkeit encodiert wird. Ähnliche Aktionen haben in diesem Raum einen geringen Abstand, wohingegen unterschiedliche Aktionen einen großen Abstand in diesem Raum haben.

Weiterhin präsentieren wir einen Ansatz zur Sensordatenfusion für das Aktionserkennungsproblem auf Basis von Faltungsnetzen für Graphen. Diese generalisieren die Konzepte von Faltungsnetzen auf nicht-Euklidische Datenstrukturen. Diese Ansätze definieren derzeit den Stand der Technik durch sehr gute Klassifikationsergebnisse auf Räumlich-Temporalen Daten wie Skelettdatensequenzen, dennoch werden diese derzeit in der Literatur nicht als Basis für die Sensordatenfusion verwendet. Mit unserem Fusion-GCN Ansatz stellen wir einen Ansatz zur Sensordatenfusion auf Basis von Faltungsnetzen für Graphen vor. Dabei werden weitere Sensoren in einen Skelettgraphen auf zwei Ebenen fusioniert. Zur Fusion in der Kanaldimension werden neue Attribute an schon existierende Knoten des Graphen angehängen. Auf der räumlichen Ebene werden neue Knoten in den Graphen aufgenommen. Wir zeigen, dass der Ansatz in der Lage ist verschiedene Modalitäten, wie Merkmale von RGB Bildern oder Messungen von Inertialsensoren in einen Skelettgraphen zu fusionieren.

Des Weiteren präsentieren wir einen Ansatz zur Segmentierung von Aktionen auf Sequenzen von Skeletten. Transformer Netzarchitekturen erreichen in jüngster Zeit sehr gute Ergebnisse zur Analyse von sequenziellen Daten. Motiviert davon, formulieren wir das Aktionsegmentierungsproblem als Objekterkennungsproblem und nutzen ein Objektdetektionsnetz basierend auf einer Transformerarchitektur als Grundlage zur Segmentierung der Aktionen auf Basis der Bildrepräsentationen. Experimente für unsere Aktionserkennungsansätze werden auf öffentlichen Datensätzen mit großem Umfang ausgeführt. Unsere Aktionserkennungsansätze zum Klassifizieren, zum Klassifizieren mit einem Referenzbeispiel sowie der Fusion von verschiedenen Modalitäten stellen auf einem Teil der Datensätze den Stand der Technik dar.

Abschließend präsentieren wir einen hybriden Benchmarkingansatz zum Evaluieren von Methoden des Imitationslernens vor. Der Benchmark besteht aus einem Datensatz, Metriken und einer Integration in einen Simulator. Der Datensatz enthält RGB-D Sequenzen von Menschen, welche Bewegungen und Manipulationsaufgaben demonstrieren. Weiterhin sind Grundwahrheiten für die Pose der Hand und der Interaktionsobjekte enthalten. Diese Sequenzen dienen als Eingabe zu evaluierender Ansätzen des Imitationslernens. Zur Imitation können die Ansätze in der simulierten Umgebung ausgeführt und durch Metriken zur Beurteilung der Trajektorienqualität oder des Effekts automatisch evaluiert werden.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Challenges	2
1.3	Applications	3
1.4	Contributions	5
1.5	Methodology	8
1.6	Publications	8
1.7	Collaborations	10
1.8	Outline	11
2	Preliminaries	13
2.1	Definitions	13
2.2	Problem Description	14
2.2.1	Action Recognition	14
2.2.2	One-Shot Action Recognition	16
2.2.3	Action Segmentation	16
2.3	Performance Metrics	17
2.4	Convolutional Neural Networks	19
2.4.1	Convolutional Layer	20
2.4.2	Activation Layer	21
2.4.3	Pooling Layer	22
2.4.4	Linear Layer	23
2.4.5	Training	23
2.5	Graph Convolutional Networks	24
2.6	Metric Learning	26
2.7	Human Pose Estimation	26
2.7.1	Human Pose Estimation on Depth Images	27
2.7.2	Human Pose Estimation on RGB Images	28
2.8	Modality Fusion	29

3	Action Recognition for Various Sensors	33
3.1	Introduction	35
3.2	Related Work	36
3.3	Approach	39
3.3.1	Signal Reduction	40
3.3.2	Signal Fusion	41
3.3.3	Sparse Representation	41
3.3.4	Dense Representation	41
3.3.5	Augmentation	43
3.3.6	Architecture	44
3.3.7	Implementation	44
3.4	Experiments	45
3.4.1	Evaluation Protocols	45
3.4.2	Datasets	47
3.4.3	Results	49
3.5	Conclusion	62
4	Multi-Modal Action Recognition using GCNs	65
4.1	Introduction	65
4.2	Related Work	67
4.3	Approach	69
4.3.1	Incorporating Additional Modalities Into a Graph Model	69
4.3.2	Fusion of Skeleton Sequences and RGB Video	70
4.3.3	Fusion of Skeleton Sequences and IMU Signals	71
4.3.4	Combining Multiple Fusion Approaches	71
4.4	Experiments	73
4.4.1	Datasets	73
4.4.2	Implementation	74
4.4.3	Comparison to the State-of-the-Art	74
4.4.4	Ablation Study	76
4.4.5	Limitations and Discussion	80
4.5	Conclusion	80
5	One-Shot Action Recognition	83
5.1	Introduction	83
5.2	Related Work	86
5.3	Signal-Level Deep Metric Learning	90
5.3.1	Problem Formulation	91
5.3.2	Representations	91
5.3.3	Feature Extraction	92
5.3.4	Metric Learning	92

5.4	Skeleton-Based Deep Metric Learning	93
5.4.1	Problem Formulation	93
5.4.2	Skeleton-DML Representation	94
5.4.3	Feature Extraction	95
5.4.4	Metric Learning	96
5.5	Experiments	97
5.5.1	Implementation	98
5.5.2	Datasets	98
5.5.3	Signal-Level Deep Metric Learning Experiments	100
5.5.4	Skeleton-DML Experiments	106
5.6	Conclusion	112
6	Action Segmentation	115
6.1	Introduction	115
6.2	Related Work	117
6.3	Approach	118
6.3.1	Detection Transformer - DEtection TRansformer (DETR) . . .	118
6.3.2	Representation	119
6.4	Experiments	121
6.4.1	PKU Multi-Modality Dataset - PKU-MMD	121
6.4.2	Implementation	121
6.4.3	Results	122
6.5	Conclusion	127
7	Benchmarking for Imitation Learning	129
7.1	Introduction	129
7.2	Related Work	132
7.3	Dataset	134
7.3.1	Setup	135
7.3.2	Testbed	135
7.3.3	Calibration	136
7.3.4	Human-Object Interactions	140
7.3.5	Sequences	140
7.4	Benchmark	141
7.4.1	Effect	144
7.4.2	Trajectory Error	145
7.4.3	Baseline	146
7.5	Conclusion	147
8	Conclusion and Outlook	149

A Simitate Sequence Examples	155
B Simitate Testbed Layout	161
C Curriculum Vitae	163
List of Tables	170
List of Figures	173
Acronyms	176
Publications	177
Bibliography	183

Chapter 1

Introduction

We are surrounded by sensors nowadays, raising the question: How can we use this data? Sensors enable great possibilities for the analysis and information extraction of its data. This question is exciting as sensors are distributed in various domains allowing a broad range of applications, from elderly care, autonomous driving, and surveillance to service robotics. Humans can learn from observations through their sensory input, which leads to whether autonomous systems can also learn new behaviors by observation from those sensors. Machine learning research made considerable progress and continues to influence many research areas that directly benefit from many advances. Still, there remain challenges, for example, in terms of generalization. Many proposed approaches aim to optimize for single tasks on single sensors. A current research problem is to develop approaches for broader generalization, i.e., by generalizing across multiple tasks or sensor modalities. The acquisition of behaviors by observation for autonomous systems also made significant advances, guided by the progress of machine learning. Crucial for those approaches is assessing the quality of imitated behaviors to advance the imitation learning field. What is typical for many computer vision or machine learning datasets is in robotics, either limited to simulated environments or hard to reproduce and verify on a larger scale due to the systems' complexity.

1.1 Motivation

The analysis of human actions and their imitation by robotic systems intersects the three research fields: computer vision, robotics and machine learning, and enables a wide range of applications.

Action recognition describes the analysis of sequential data to estimate an action label. Most prominently, image sequences or their respective transformations into a feature space (like skeleton sequences) are considered as input for action recognition approaches. The generalization of action recognition approaches to various data modal-

ities or their fusion is favorable and enables flexible transfer from vision systems to embedded devices or mobile robots. The task of action recognition is a specialization of more general human motion analysis [AC99; AR11] and has close relations to geometric human modeling [Joh73; MN78] and body tracking [Arg+09]. Early action recognition approaches were image-based [Hog83], but later approaches transformed full video sequences into a feature space [PN94] that allows identifying different actions. The recognition of human actions in videos (image sequences) has close relations to more general video classification.

Imitation learning, in the context of this thesis, describes the ability of a robotic system to imitate an observed human activity from sequential data. Thus, it extends the estimation of an action class in a sequence towards translating the observed activity into a robotic behavior. A key research problem forms the imitation of observations that can be gathered from images directly.

Like many other research fields, both research areas benefited highly from the advances of deep neural networks in the last decade. Hand-crafted feature descriptions have been replaced by learned feature representations, which have been demonstrated to generalize across many domains like image classification [KSH12; He+16], speech recognition [PMC15] and even natural language processing [CW08; SZ14; ZZL15]. In our case, the motivation of this thesis also relies on observation of limitations in current service robotic systems as we experienced them throughout the participation in robot competition attendances [MSP17; Mem+18c; Mem+18b]. We identified shortcomings in two research domains that we tackle throughout this thesis: We found a lack of approaches that generalize well across various sensor modalities for the action recognition task. Further, we found a lack of comparability in the evaluation of imitation learning approaches.

1.2 Challenges

Challenges for the recognition of actions are introduced by the wide range of different applications, their different sensor modalities, and setup variations. Sensor modalities range from Inertial Measurement Units (IMUs), Wi-Fi Channel State Information (CSI) fingerprints, motion capturing systems, image sequences from cameras and Global Positioning System (GPS). Some sensors suitable for the action recognition task are depicted in Fig. 1.1. Sensor position setups, e.g., for cameras differ (like depicted in Fig. 1.2) between static external observations (Fig. 1.2a) and moving positions like for first person views (Fig. 1.2b) or mounted on mobile platforms like UAVs (Fig. 1.2c). Approaches that generalize well across different setups are favorable, as they allow for more flexible applications. Popular evaluation protocols in action recognition datasets are used to evaluate the cross setup capabilities by presenting different views of a sequence. Besides *cross-setup* evaluation protocols, *cross-subject* protocols yield interesting insights

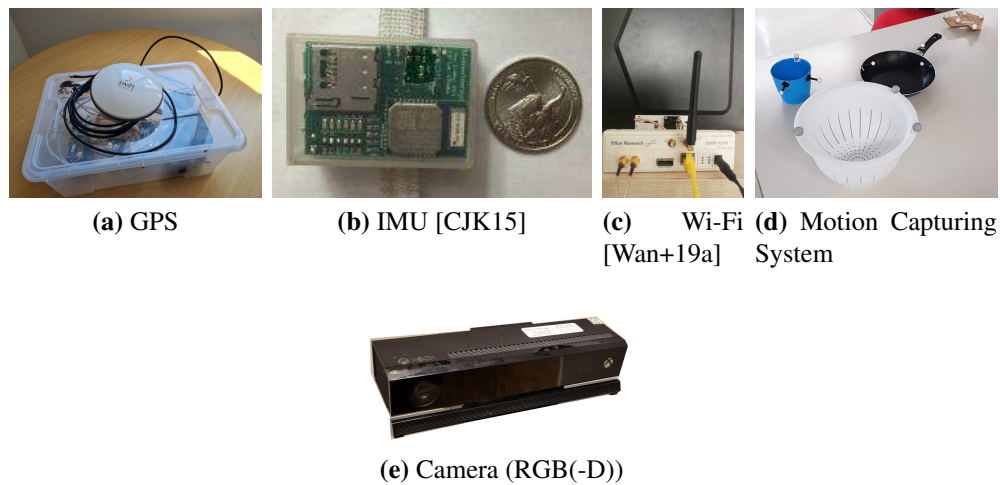


Figure 1.1: An excerpt of sensors that allow action recognition.

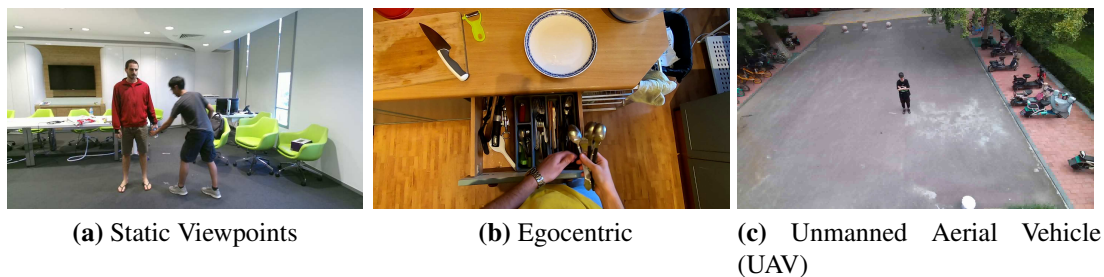


Figure 1.2: The image on the left shows a static camera position from the NTU RGB+D 120 [Liu+20a] dataset. The center image shows an egocentric view from the EPIC-KITCHENS dataset [Dam+21]. The right image shows an aerial view from the UAV-Human dataset [Li+21].

about how good an approach can generalize between different persons that have not been seen during training time and might execute the same actions quite different. Advances from the action recognition research can be used to advance the imitation learning field, such that robots can be enabled to learn by observations if they are able to understand them. In imitation learning research, a key challenge is to imitate behaviors that has been observed from image streams.

1.3 Applications

Action recognition approaches are applicable in various domains, ranging from Human Robot Interaction (HRI) to surveillance. Especially, approaches that can handle video streams as input enable wide applications, as can be seen in Fig. 1.3, where multiple

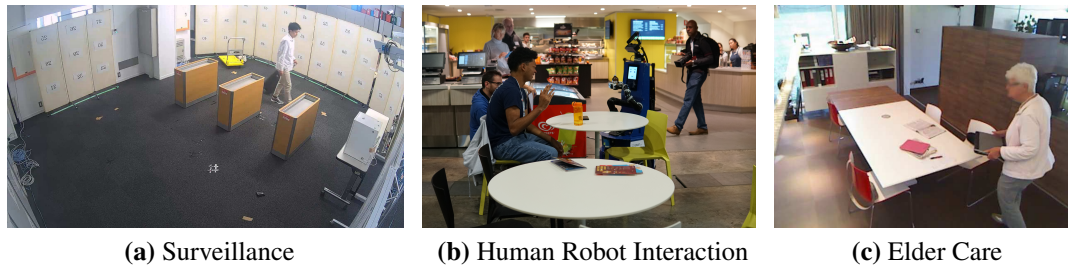


Figure 1.3: Three application settings that demonstrate sample applications. 1.3a shows a surveillance setting from the MMAAct dataset [Kon+19], 1.3b shows a possible application from our RoboCup@Home participation with our service robot LISA where the robot should find a waving guest in an unknown restaurant. 1.3c gives an example for a possible elder care setting from the Toyota Smarthome dataset [Das+19].

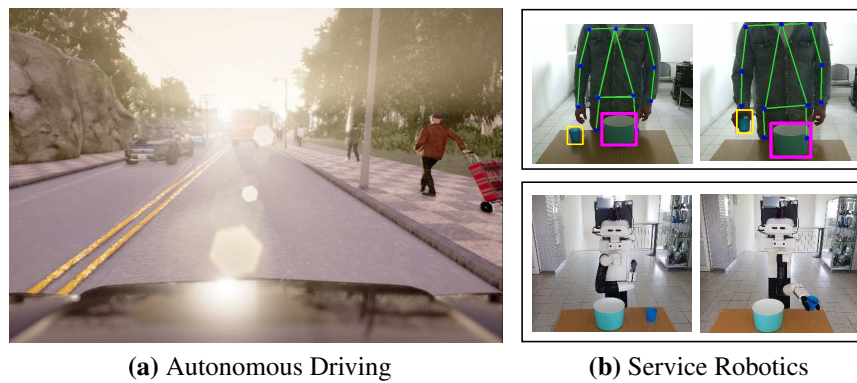


Figure 1.4: Imitation learning applications from various domains like autonomous driving (a) [Dos+17] or service robotics (b).

applications from various domains are given. Benchmarks and datasets [Hei+15; CZ17; Kon+19; Dam+21], often accompanied by workshops at major computer vision conferences, have been formed to address shortcomings in current approaches and like the ActivityNet [Hei+15] have been built around various action recognition problems. A large set of publicly available datasets for various applications, sensor setups and levels of abstraction have been built. In Fig. 1.4 we give example applications for imitation learning approaches for autonomous driving and service robotics.

Surveillance Approaches based on action recognition in video streams [Niu+04; DG07; Han+18; Ull+19] can be used to detect security issues like aggressive behavior, theft, or destruction. In a surveillance setting, cameras are usually mounted at a ceiling level and have a wide field of view to observe critical locations. Action prediction could potentially even prevent criminal acts [KF18]. On a broader

view, these systems can also be used to widely track the activities of humans and analyze behaviors and create individual activity profiles for marketing or social classifications. Action recognition and prediction arise many ethical questions should be seen critical depending on their application and consequent decisions.

Elder Care Health and security monitoring in senior homes, often referred to as assisted living or ambient assistive living [Che+13] allows for another wide range of applications. Senior people in domestic smart homes to call for assistance in case of emergency forms an interesting application with broader impact. For instance, triggering an emergency call in case of a fallen person [Nou+07; ST17] could potentially save someone's life. More general activity analysis for senior people, especially in rapidly aging population in developed countries, allows health monitoring [NGC15]. Irregularities in the medication can potentially be detected and the senior person or caregivers can be notified. Great datasets, in realistic settings have been proposed, especially in a smart home with senior people [Das+19].

Human Robot Interaction In applied robotics, a natural interaction between humans and robots is crucial for acceptance and also allows for wide applications ranging from industrial applications [AH15] to social robotics. Natural human-robot-interaction requires real-time action recognition capabilities [Son+20b; Fan+17a]. Planning social acceptable and safe trajectories based on observed actions in crowded settings can improve the navigation of robots in domestic environments [CKG16a]. Whole conferences like RO-MAN and HRI target social robotics research and focus on human-robot-interaction, where the recognition of actions performed by humans play a vital role.

Autonomous Driving To relieve expert programmers for autonomous systems, imitation learning approaches are interesting candidates, that become especially favorable when dealing with large-scale datasets. Widely spread is imitation learning for applications in autonomous driving [Keb+20; Pan+20]. Of increasing research interest are also imitation learning approaches for service robots [Fan+19]. Our action recognition approaches are also candidates for application in autonomous driving for driver-behavior analysis [Zha+17] e.g., to detect driver sleepiness [Bac+20]. Our one-shot action recognition approaches may even be suitable for anomaly detection on various sensor modalities.

1.4 Contributions

In this thesis, we present novel approaches for the recognition and imitation of activities. Our approaches are applicable for various sensors applications and setups that were mentioned in Section 1.2. The presented approaches allow applications for

- multimodal action recognition,
- the multimodal action recognition given a single reference sequence,
- the segmentation of actions from continuous streams,
- the transfer from sensor data into a simulative environment to train robot behaviors,
- assessing the quality of human actions imitated by robots.

In detail the contributions can be categorized and summarized as follows:

Action Recognition for Various Data-Modalities We presented approaches for action recognition on various data modalities like skeleton sequences, inertial measurement units, motion capture data or Wi-Fi CSI fingerprints. Sequential data originating from different sensors are transformed to a common image representation. A convolutional neural network is trained to recognize activities from the representations. An approach proposing a common sparse representation for various data modalities was presented [MTP20a]. Further sensor data fusion methods and a filtering method to prevent the representation from overloading were presented. A dense representation unifying various data modalities is presented in this thesis. The approaches were evaluated on publicly available datasets with four different data modalities and up to 400 different action classes.

Multimodal Action Recognition We presented an approach for the fusion of sensor modalities based on a hierarchical pose graph. Additional sensor modalities are incorporated into the graph representation either on a channel dimension, by introducing additional node attributes, or a spatial dimension, by introducing new nodes into the graph. Flexible modality fusion for the fusion of accelerometer, gyroscope and orientation sensors are demonstrated in an early fusion setting on a representation level. The Fusion-GCN approach was presented in [DMP21].

One-Shot Action Recognition for Various Data-Modalities In contrast to the action recognition problem, where a model is trained to recognize known actions in various setups, the one-shot action recognition problem has to recognize previously unseen actions with a single reference sample. We proposed metric learning-based approaches based on the previously mentioned representations. Instead of learning a model that predicts an action class, the metric learning approach learns a model to transform action representations into an embedding space in which low distances reflect high action similarity and high distances reflect dissimilar actions. A signal level approach for various data-modalities and their fusion was presented in [MTP20b]. In [Mem+22] a focus on skeleton-based one-shot action

recognition was set with an extensive evaluation on various related representations.

Action Segmentation Associating action class labels and start- and end-times to untrimmed-sensor data streams like videos or skeleton sequences is a highly practical task and the basis for many applications. The action segmentation task of continuous sensor-data streams has close relation to the action recognition task, which commonly limits to trimmed sequences. We present an action segmentation approach for the segmentation of skeleton sequences based on transformer networks [HMP21]. Like for our previous approaches [MTP20a; MTP20b; Mem+22], we represent skeleton sequences in an image. Then we propose to use apply a transformer-based object-detection approach to segment the skeleton sequences.

Benchmarking of Imitation Learning We presented a benchmark to quantitatively assess the performance of imitation learning approaches [Mem+19a]. With the benchmark, we propose to integrate real sensor measurements into a simulated environment (Real-to-Sim). This is achieved by the calibration of an RGB-D camera against a motion capturing system. 1938 sequences have been recorded. The idea is that sequences of demonstrations serve as input to imitation learning approaches that learn a policy to imitate the movement or behavior. An effect metric and trajectory metric are proposed to assess the imitation performance of a robotic system in the simulated environment.

To foster reproducible research for verification, reproduction, and extension of our results, we provide the source code, datasets, and models for the core contributions of this thesis. Our Convolutional Neural Network (CNN)-based action recognition approach [MTP20a] is available on GitHub¹. Our Fusion-GCN approach for incorporation of various sensor data-modalities into a skeleton-graph with Graph Convolutional Networks (GCNs) is available². Our one-shot action recognition approaches for multi-modal action recognition on a signal-level formulation [MTP20b]³ and with a focus on skeleton sequences [Mem+22]⁴. The Simitate benchmarking environment and dataset are available on a dedicated project page⁵.

Special attention for the selection of the depending libraries is also on open source that allow royalty-free and unlimited reproduction. For better comparability, we conducted our experiments on public datasets.

¹https://github.com/raphaelmemmesheimer/gimme_signals_action_recognition

²<https://github.com/mduhme/fusion-gcn>

³<https://github.com/raphaelmemmesheimer/sl-dml>

⁴<https://github.com/raphaelmemmesheimer/skeleton-dml>

⁵<https://agas.uni-koblenz.de/simitate>

1.5 Methodology

All approaches, that are presented in this thesis, follow quantitative evaluation to assess their performance. Commonly used performance metrics and the evaluation on publicly available datasets ensure better comparability and reproduction. Further, we introduce a custom benchmark for imitation learning which introduces a novel dataset and metrics for the evaluation of the imitation. These metrics are inspired by the metrics used for the evaluation of mapping and pose estimation approaches [Küm+09; Stu+12; GLU12].

1.6 Publications

Parts of this thesis have been previously published in peer-reviewed, international, conference proceedings (chronological order):

- **Raphael Memmesheimer**, Ivanna Kramer, Viktor Seib, and Dietrich Paulus. “Simitate: A Hybrid Imitation Learning Benchmark”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2019, Macau, SAR, China, November 3-8, 2019*. IEEE, 2019, pp. 5243–5249. DOI: 10.1109/IROS40897.2019.8968029
- **Raphael Memmesheimer**, Nick Theisen, and Dietrich Paulus. “Gimme Signals: Discriminative signal encoding for multimodal activity recognition”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January 24, 2021*. IEEE, 2020, pp. 10394–10401. DOI: 10.1109/IROS45743.2020.9341699
- **Raphael Memmesheimer**, Nick Theisen, and Dietrich Paulus. “SL-DML: Signal Level Deep Metric Learning for Multimodal One-Shot Action Recognition”. In: *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*. IEEE, 2020, pp. 4573–4580. DOI: 10.1109/ICPR48806.2021.9413336
- Simon Häring, **Raphael Memmesheimer**, and Dietrich Paulus. “Action Segmentation on Representations of Skeleton Sequences Using Transformer Networks”. In: *2021 IEEE International Conference on Image Processing, ICIP 2021, Anchorage, AK, USA, September 19-22, 2021*. IEEE, 2021, pp. 3053–3057. DOI: 10.1109/ICIP42928.2021.9506687
- Michael Duhme, **Raphael Memmesheimer**, and Dietrich Paulus. “Fusion-GCN: Multimodal Action Recognition Using Graph Convolutional Networks”. In: *Pattern Recognition - 43rd DAGM German Conference, DAGM GCPR 2021, Bonn,*

Germany, September 28 - October 1, 2021, *Proceedings*. Ed. by Christian Bauckhage, Juergen Gall, and Alexander G. Schwing. Vol. 13024. Lecture Notes in Computer Science. Springer, 2021, pp. 265–281. DOI: 10.1007/978-3-030-92659-5_17

- **Raphael Memmesheimer**, Simon Häring, Nick Theisen, and Dietrich Paulus. “Skeleton-DML: Deep Metric Learning for Skeleton-Based One-Shot Action Recognition”. In: *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022, Waikoloa, HI, USA, January 3-8, 2022*. IEEE, 2022, pp. 837–845. DOI: 10.1109/WACV51458.2022.00091

Video abstracts of the core contributions of this thesis have been created in order to make the research topics more accessible. The video abstracts are shown and referenced in Fig. 1.5.

Closely related to the content of this thesis are also the following publications that have been published during the writing of this thesis:

- **Raphael Memmesheimer**, Ivanna Mykhalchyshyna, and Dietrich Paulus. “Gesture Recognition On Human Pose Features Of Single Images”. In: *9th IEEE International Conference on Intelligent Systems, IS 2018, Funchal, Madeira, Portugal, September 25-27, 2018*. Ed. by Ricardo Jardim-Gonçalves, João Pedro Mendonça, Vladimir Jotsov, Maria Marques, João Martins, and Robert E. Bierwolf. IEEE, 2018, pp. 813–819. DOI: 10.1109/IS.2018.8710515
- **Raphael Memmesheimer**, Viktor Seib, and Dietrich Paulus. “homer@UniKoblenz: Winning Team of the RoboCup@Home Open Platform League 2017”. In: *RoboCup 2017: Robot World Cup XXI [Nagoya, Japan, July 27-31, 2017]*. Ed. by Hidehisa Akiyama, Oliver Obst, Claude Sammut, and Flavio Tonidandel. Vol. 11175. Lecture Notes in Computer Science. Springer, 2017, pp. 509–520. DOI: 10.1007/978-3-030-00308-1_42
- **Raphael Memmesheimer**, Ivanna Mykhalchyshyna, Viktor Seib, Tobias Evers, and Dietrich Paulus. “homer@UniKoblenz: Winning Team of the RoboCup@Home Open Platform League 2018”. In: *RoboCup 2018: Robot World Cup XXII [Montreal, QC, Canada, June 18-22, 2018]*. Ed. by Dirk Holz, Katie Genter, Maarouf Saad, and Oskar von Stryk. Vol. 11374. Lecture Notes in Computer Science. Springer, 2018, pp. 512–523. DOI: 10.1007/978-3-030-27544-0_42
- Pascal Schneider, **Raphael Memmesheimer**, Ivanna Kramer, and Dietrich Paulus. “Gesture Recognition in RGB Videos Using Human Body Keypoints and Dynamic Time Warping”. In: *RoboCup 2019: Robot World Cup XXIII [Sydney, NSW, Australia, July 8, 2019]*. Ed. by Stephan K. Chalup, Tim Niemüller, Jackrit

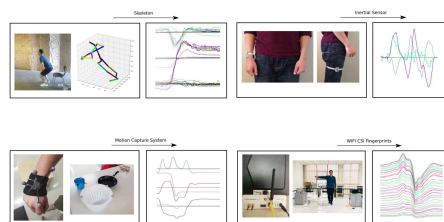
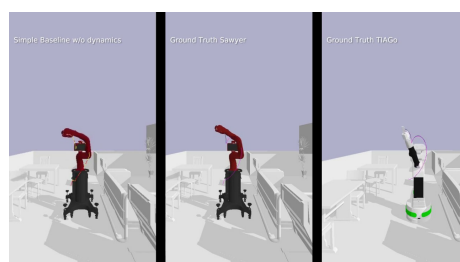
Suthakorn, and Mary-Anne Williams. Vol. 11531. Lecture Notes in Computer Science. Springer, 2019, pp. 281–293. DOI: 10.1007/978-3-030-35699-6_22

- Ivanna Kramer, Niko Schmidt, **Raphael Memmesheimer**, and Dietrich Paulus. “Evaluation Of Physical Therapy Through Analysis Of Depth Images”. In: *28th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2019, New Delhi, India, October 14-18, 2019*. IEEE, 2019, pp. 1–6. DOI: 10.1109/RO-MAN46459.2019.8956435
- **Raphael Memmesheimer**, Viktor Seib, Tobias Evers, Daniel Müller, and Dietrich Paulus. “Adaptive Learning Methods for Autonomous Mobile Manipulation in RoboCup@Home”. In: *RoboCup 2019: Robot World Cup XXIII [Sydney, NSW, Australia, July 8, 2019]*. Ed. by Stephan K. Chalup, Tim Niemüller, Jackrit Suthakorn, and Mary-Anne Williams. Vol. 11531. Lecture Notes in Computer Science. Springer, 2019, pp. 565–577. DOI: 10.1007/978-3-030-35699-6_46
- **Raphael Memmesheimer**, Ivanna Kramer, Viktor Seib, Nick Theisen, and Dietrich Paulus. “Robotic Imitation by Markerless Visual Observation and Semantic Associations”. In: *2020 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2020, Ponta Delgada, Portugal, April 15-17, 2020*. IEEE, 2020, pp. 275–280. DOI: 10.1109/ICARSC49921.2020.9096123

At the beginning of non-first-authorship chapters, we give details about the own contribution. A complete list of publications is given in the Publications appendix.

1.7 Collaborations

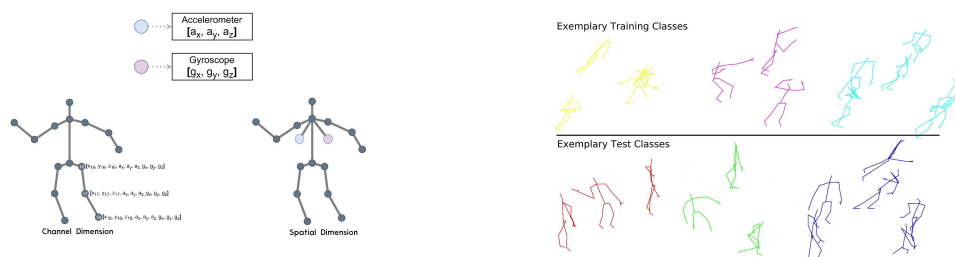
During my PhD studies, I supervised various research projects as well as bachelor- and master theses. Some of them have been extended and published afterwards. The approach for segmenting activities from skeleton streams using transformer networks [HMP21], presented in Chapter 6, is based on the master thesis of Simon Häring. The implemented representations for his master thesis allowed the in-depth experiments of the extensive representation comparison in a one-shot action recognition setting [Mem+22]. The *Fusion-GCN* approach [DMP21], presented in Chapter 4 was extended with Michael Duhme based on his master thesis. Inspiration, that later led to the signal-level representations for the action recognition tasks on various sensors, has been gathered from the research project and collaboration with Pascal Schneider



<https://youtu.be/usmmDaFREC4> https://youtu.be/oDatim_nJEg



https://youtu.be/Wdy_YPPiYgc <https://youtu.be/yNEuzqzNBSc>



<https://youtu.be/CriyQgqCTrs> <https://youtu.be/jH5eMDZfMyY>

Figure 1.5: Video abstracts of the contributions of this thesis (in chronological order).

[Sch+19]. Of great assistance for the acquisition of the data for the Simitate, imitation learning benchmark [Mem+19a], presented in Chapter 7, was the access to IS-RoboNet@Home Test Bed of Institute for Systems and Robotics at the Instituto Superior Técnico, U.Lisboa in Portugal. Further, for the Simitate benchmark [Mem+19a], Ivanna Kramer, was of great support during the data acquisition and discussions. Together with Nick Theisen most of the problem formulations [MTP20a; MTP20b; Mem+22], were simplified and improved.

1.8 Outline

We introduce preliminaries that are shared across the thesis in the following Chapter 2. Chapters 3 - 7 introduce our research contributions and are mostly self-contained with

their individual related work discussion and conclusion. Chapter 3, presents action recognition methods using CNNs for various sensor modalities and their fusion in a *supervised* training setting. In Chapter 4, we focus on a method for the fusion of various sensor modalities into a skeleton-graph based on GCN. Approaches framed in a *semi-supervised* setting are presented in Chapter 5. The segmentation of actions on skeleton sequences with a transformer network is presented in Chapter 6. After presenting methods to recognize and segment human performed actions, we present a benchmarking method for evaluating imitation learning approaches in a robotics context in Chapter 7. Finally, Chapter 8 provides a general conclusion.

Chapter 2

Preliminaries

In this chapter, we present the foundation that is used in multiple later chapters. We give the fundamental definitions, problem descriptions and performance metrics. After, we introduce the fundamentals of a Convolutional Neural Network (CNN) and GCNs. Finally, we introduce related approaches for human pose estimation on Red Green Blue (RGB) and Red Green Blue Depth (RGB-D) sequences that we use fundamentally among our approaches.

2.1 Definitions

Throughout the thesis, we want to establish common definitions, as many of the terms used in research get alternating definitions.

In three of the following chapters, we propose methods related to the action recognition problem. Different definitions for an action are existing. In this thesis, we follow a definition of *action*, which is based on a combination of the *activity* and *action* definitions by Bobick [Bob97]. We soften the granularity of the definition here, as our approaches do not distinguish in the applied methods for handling them. Large-scale datasets also commonly consist of activities and actions without distinguishing between them explicitly. An action is then defined as follows:

Definition 1: Action

An action ranges from sequences of motion towards larger-scale events, which typically include interaction with the environment and causal relationships.

This definition includes short human motion sequences, sometimes also referred to as gesture, as well as more complex human motion sequences like interactions with appliances or the environment and interactions between two or more persons. Actions can range from short sequences to more enduring sequences.

For our action recognition related tasks, we propose approaches that aim at generalizing well across different sensor modalities. Many definitions for the abstract term of a signal have been developed in various domains. We follow the definition by [Opp+97; Cha18] of signals as:

Definition 2: Signal

A signal is a function of one or more independent variables that contain information about the behavior or nature of some phenomenon.

Examples of signals range from acoustic sound waves that are converted to voltages by microphones, orientation, and velocities measured by a gyroscope or event images as captured by image sensors, and also covers transformations from images to higher level features like human pose features.

2.2 Problem Description

The focus of this thesis is on action recognition on various sensor data modalities, which we interpret as a classification task. Our proposed approaches aim at a general formalization by defining multivariate signal streams as the standard input for our action recognition methods. Signal streams originate either directly from sensors like inertial measurement systems, marker guided position estimation devices like motion capturing systems, Wi-Fi CSI-fingerprints or skeleton sequences. Throughout this thesis, we observe skeleton sequences with a special consideration because of the large-scale public available datasets and their general applicability. Skeleton-sequences can be gathered by transforming input images to human pose features [Cao+21; Sho+11] from passively observing sensors.

Fig. 2.1 gives a categorization of action recognition approaches. *Action Recognition* infers a single action class for a given sequence. *Action Segmentation* detects multiple action chunks, formed by start- and end-times, as well as their corresponding class label. In case of multi-person actions, a single class label is inferred for a chunk, in our case the most active person. *Spatio-Temporal Action Localization*, in contrast, infers spatio-temporal information about the humans, and their actions. In addition, a bounding box per person containing the area of interest and an action label for each frame is predicted. An additional person identifier, that tracks persons throughout a sequence, is favorable.

2.2.1 Action Recognition

In machine learning, given an optimization algorithm, a loss function, a model and a dataset [GBC16], a model can be trained to solve for a task. One of the most fundamen-

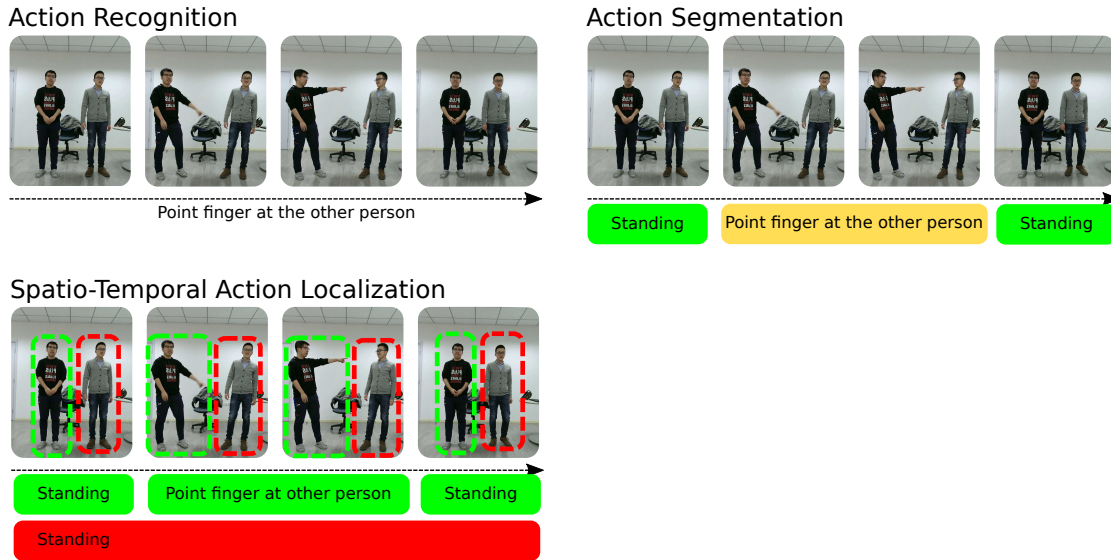


Figure 2.1: Classification of action recognition approaches.

tal tasks is the classification of data. For that, we want a function that maps an input to an label $l \in \{1, \dots, C\}$

$$f : \mathbb{R}^n \rightarrow l,$$

that transforms an n dimensional input vector of data to numeric class identities. In machine learning the function f , to transform the input data to associate class labels, is learned. During inference, the learned function that is related to as a model is used to predict a class label:

$$\mathbf{y} = g(\mathbf{x}),$$

where \mathbf{x} is an input vector and \mathbf{y} is a class identity encoded in an one-hot vector.

Throughout this thesis, we transform motion sequences into images and use them as basis for our classification approaches. In that case, the input is assumed to be an image \mathbf{I} , which is represented as a matrix $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$. The problem then becomes finding a function that associates a class label to an image:

$$f : \mathbf{I} \rightarrow l$$

in a final model that predicts an action class:

$$\mathbf{y} = g(\mathbf{I}).$$

We consider the standard action recognition task as a supervised learning task for which a model is trained on C classes, where the training and test sets share the same C classes. Thus, a test set \mathcal{T} shares the same classes as the training set \mathcal{D} in contrast to the one-shot action recognition problem that is introduced in the following section.

2.2.2 One-Shot Action Recognition

For one-shot action recognition [Liu+20a], the goal is similar to the action recognition problem. Action class labels should be estimated, but the amount of training samples of a specific class to recognize is limited to a single reference sample. That yields a more flexible and practical application formulation in novel scenarios. In contrast to the standard action recognition setting as it was presented above, in a one-shot action recognition setting C classes are known in a training set \mathcal{D} , while the test set \mathcal{T} contains U novel classes, providing a single reference sample per class in an auxiliary set \mathcal{A} , where $|\mathcal{A}| = U$. This constraint doesn't allow training a model that predicts a known class label but learns a more generic model that transforms an input image to an embedding space, preferably reflecting semantic relevance, that allows class association of the test samples to the classes by finding the nearest neighbor in the embedding space.

In the one shot action recognition setting, we therefore want to find a transformation into an embedding space:

$$g : \mathbf{I} \rightarrow \mathbb{R}^n,$$

that allows associating class labels to reference actions $u \in \{1, \dots, U\}$ by finding the nearest neighbor:

$$f : \mathbb{R}^n \rightarrow u.$$

For the transformation to the embedding space a metric learning approach can be utilized.

2.2.3 Action Segmentation

The action segmentation task tackles the problem of temporally segmenting sequences. An action segment is described with an action class and start and end frames encoding the duration of the action. An encoded sequence can contain multiple such tuples. Fig. 2.1 visualizes on the top right an example of the action segmentation problem. As for the previous two problem descriptions, we assume an image representation, encoding signals as input and target the estimation of n action segments in the underlying representation. Formally, the action segmentation can be described with a function that transforms an image representation to a set of action segments as follows:

$$h : \mathbf{I} \rightarrow \{(t_{\text{start}}, t_{\text{end}}, l)_0, \dots, (t_{\text{start}}, t_{\text{end}}, l)_n\},$$

where t_{start} and t_{end} are the timestamps for the start and end of the segment and l corresponds to the estimated segment label. Multiple segments can be contained in an image representation.

2.3 Performance Metrics

Relying on common performance metrics for the evaluation of an approach is as important as the usage of publicly available datasets. In later chapters, we present approaches for the action recognition task and temporal action segmentation. For better comparability, we report performance metrics that are used for the evaluation of our approaches. A good reference to various performance metrics is given by Manning et al. [MRS08]. Metrics for the evaluation of action recognition task, which is considered to be a classification task with balanced and imbalanced datasets, are introduced. Further, we present a metrics for the performance of temporal action segmentation approaches. For the evaluation of imitation learning approaches, we propose new metrics for the evaluation focusing on the effect in Section 7.4.1 and the evaluation of the trajectory in Section 7.4.2 as part of our Simitate benchmark.

The *Accuracy* [] is the standard metric for classification tasks with balanced classes and is defined as the proportion of correct predictions to the total number of predictions:

$$\text{Accuracy} = \frac{\#\text{Correct Predictions}}{\#\text{Total Predictions}}$$

or

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},$$

where TP, TN denote true positives and true negatives, likewise, the FP, FN denote false positives and false negatives. The accuracy estimates how good the classification performance is on the given set, and therefore is simple to interpret and calculate. However, on largely unbalanced datasets, the accuracy is biased towards more frequent sample classes, therefore metrics to handle class imbalances like the *Mean Per-Class Accuracy (mpcA)* for which accuracies per class are calculated and afterwards the mean over all accuracies is taken.

The F_1 -*measure* is defined as the harmonic mean of precision and recall. If either precision or recall is much lower than the other, the F_1 measure will significantly lean towards the lower value.

Given the amount of true positives and false positives, we can estimate the fraction of the correct estimates as precision with:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$

and the fraction of overall correct segments recall as:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

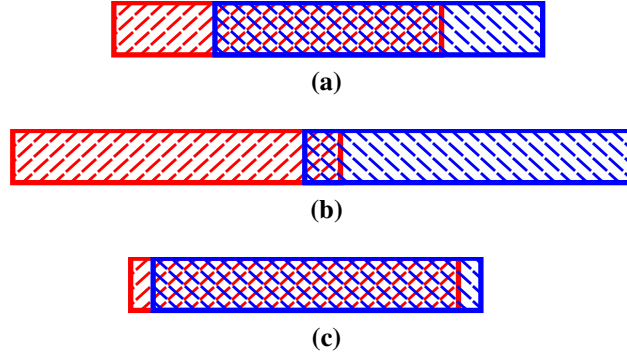


Figure 2.2: Intersection over Union (IoU) visualization. $A_{\text{Groundtruth}}$ depicting the ground-truth segment and $A_{\text{Predicted}}$ depicting the estimated segment. Different occurring examples are given in (a), (b), (c). The crossed segment depicts the overlap between the ground-truth and the estimate. An example for a bad estimate that falls below a threshold of ω is given in (b) and therefore considered to be a false positive. A good estimate is shown in (c).

Depending on the system-requirements, a system might optimize for higher precision or higher recall. To give a better overall performance estimate, the F_1 -measure combines precision and recall in a single metric with:

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}.$$

The F_1 -measure is high when both precision and recall are high and has benefits for class imbalanced datasets.

The *Mean Average Precision (mAP)* is a metric that is commonly used to evaluate object detection methods, but also translates well to action segmentation tasks. We follow the derivation by Manning et al. [MRS08, p. 158–160] and Liu et al. [Liu+17a]. First, the *Intersection over Union (IoU)* of two segments $A_{\text{Groundtruth}}$ as the ground-truth segment and $A_{\text{Predicted}}$ as the predicted segment. The IoU is then calculated as follows:

$$\text{IoU} = \frac{|A_{\text{Groundtruth}} \cap A_{\text{Predicted}}|}{|A_{\text{Groundtruth}} \cup A_{\text{Predicted}}|}.$$

If the IoU is above a threshold ω :

$$\text{IoU} > \omega,$$

the predicted segment has a high overlap and is considered to be a true positive. Analog for the false positives if the IoU is below a threshold. Fig. 2.2 visualizes two segments and overlap. The higher the overlap, the higher the IoU. Extensions of the IoU, like the Generalized Intersection over Union (GIoU) [Rez+19] reward also non-overlapping bounding boxes and therefore are more suitable as for the integration into a bounding-box regression learning method, but also can be used as a metric.

To gather the average precision per class, we calculate the precision and recall over all sequences K and rank them by their confidence scores.

The interpolated precision $\text{Precision}_{\text{interp}}$ is defined as the highest precision for any recall level $\text{Recall}' \geq \text{Recall}$:

$$\text{Precision}_{\text{interp}} = \max_{\text{Recall}' \geq \text{Recall}} \text{Precision}(\text{Recall}').$$

The average precision is the area below the sorted precision-recall curve and given by:

$$\text{AP} = \int_0^1 \text{Precision}_{\text{interp}}(\text{Recall}) d\text{Recall},$$

which is practically achieved by discretizing the integration by a summation:

$$\text{mAP}(\mathcal{Q}) = \frac{1}{|\mathcal{Q}|} \sum_{j=1}^{|\mathcal{Q}|} \underbrace{\frac{1}{m_j} \sum_{k=1}^{m_j} \text{Precision}_{\text{interp}}(\text{Recall}_{jk})}_{\text{AP}},$$

where m_j is the number of action occurrences $\{d_1, \dots, d_{m_j}\}$ for parts of a retrieval set \mathcal{Q} and Recall_{jk} is the recall result of a ranked k retrieval result [Liu+17a]. For the evaluation \mathcal{Q} is defined over all actions and videos separately denoted as $\text{mAP}_{\text{action}}$ and $\text{mAP}_{\text{video}}$.

The mAP is used to evaluate our action segmentation approaches. In practice, we use the proposed evaluation scripts of the PKU-MMD [Liu+17a] dataset for better comparability.

2.4 Convolutional Neural Networks

The wide use of Convolutional Neural Networks CNNs lead to a paradigm shift, from previously often handcrafted feature designs to learned features and their combinations. CNNs form the basis of large parts of the methods described throughout this thesis. Therefore, we introduce the different layers that a CNN consists of. CNNs are widely used in image analysis but can be transferred to a wide set of tasks from various domains.

The central element of CNNs is the convolutional layer, which consists of filters, that, during a training phase, learn to identify patterns. With each layer the patterns that can be identified get more complex.

An overview of a CNN is given in Fig. 2.3. The actual learned elements are the filters denoted in red.

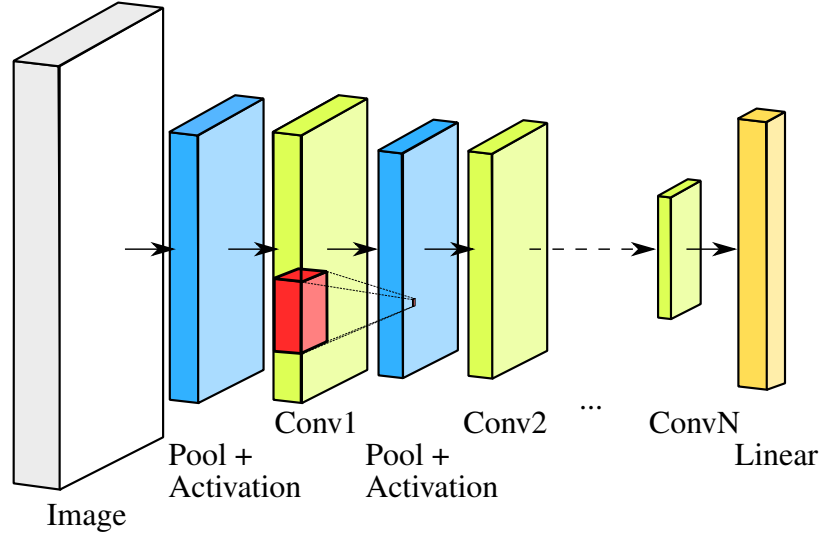


Figure 2.3: An exemplary CNN overview. The input image on the right, the convolutional layers in green and the linear layers that flatten the results after several convolutions to a feature vector. Exemplary filter weights are denoted in red.

2.4.1 Convolutional Layer

The central element of a CNN is a convolutional layer, which performs an element wise multiplication with a kernel \mathbf{K} for each position at the input image \mathbf{I} . We follow the definition of Goodfellow et al. [GBC16] for a convolution operation at position i, j over valid positions m, n :

$$\text{conv}(i, j) = (\mathbf{I} * \mathbf{K})(i, j) = \sum_m \sum_n \mathbf{I}(i - m, j - n) \mathbf{K}(m, n).$$

Many libraries implement a convolution as a cross-correlation, which is defined as:

$$\text{conv}(i, j) = (\mathbf{I} * \mathbf{K})(i, j) = \sum_m \sum_n \mathbf{I}(i + m, j + n) \mathbf{K}(m, n),$$

with a flipped kernel. As the weights for the kernels will be adjusted throughout the training process, the cross-correlation will just learn a flipped kernel. Thus, both operations result in the same weights.

Fig. 2.4 gives an exemplary convolutional operation. Traditionally, kernels have been handcrafted. In CNNs, the weights for the convolutional kernels are learned to optimize for a given loss function. In a CNN architecture, the number of layers, and sizes of filters are defined. More recently, the architectures are also samples from an architecture space and allows the definition of loss functions to optimize the gathered architecture e.g., towards high accuracies on the validation set or to minimize the number of floating point operations. In this thesis, we do not propose novel architectures

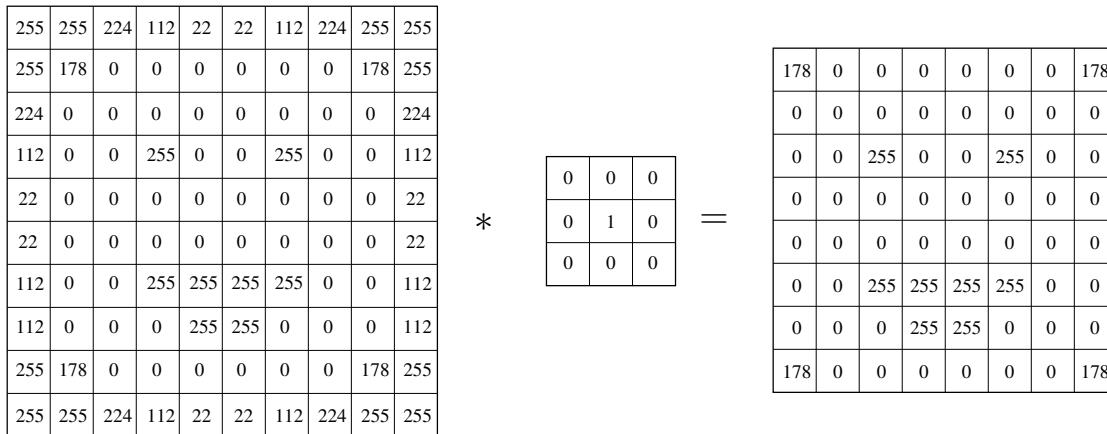


Figure 2.4: Example of a convolution operation.

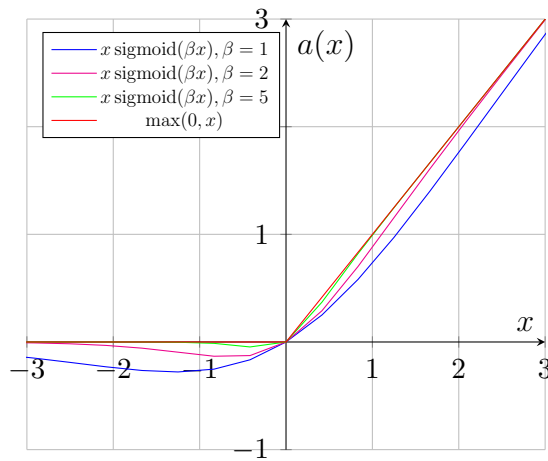


Figure 2.5: Different activation functions.

but take inspiration from well established architectures by using them as backend models for learning models to image representations like the ResNet [He+16] architecture family which allows for identity mappings (also known as skip-connections) and the EfficientNet [TL19] architecture family. The identity mappings of ResNet increased the accuracy for deeper neural networks and stabilized the training. The EfficientNet architecture family proposed to scale the number of trained filters, the filter sizes and depth of the CNN at the same time.

2.4.2 Activation Layer

An activation layer wraps an activation function that decides which neurons are activated and to which extent. Further, the activation function adds non-linearity to the training of

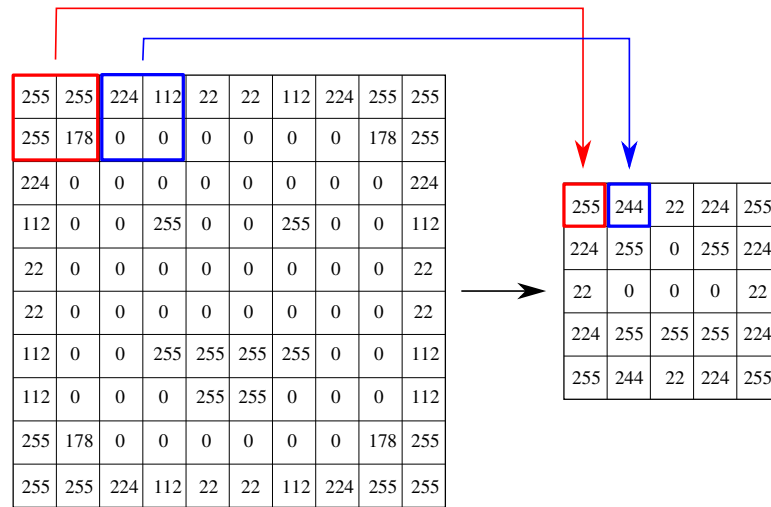


Figure 2.6: A max pool operation with (2,2).

a neural network. Historically, sigmoid and tanh [APB17; KO11] activation functions have been widely used, but alternatives have prevailed, mostly because they suffer from the vanishing gradient problem. Modern architectures often employ Rectified Linear Unit (ReLU) [NH10] which takes the element-wise max of 0 and a given input value x :

$$a(x) = \max(0, x).$$

The Rectified Linear Unit (ReLU) is gradient preserving and due to its simplicity, it is efficient to compute and superior performance over sigmoid and tanh on deeper CNNs as shown in empirical studies [Xu+15; MSM17]. More recently, the swish activation has been proposed by leveraging automatic search techniques for finding new activation functions [RZL18]. The swish activation function is defined as:

$$a(x) = x \operatorname{sigmoid}(\beta x),$$

where the sigmoid function is defined as $\operatorname{sigmoid}(z) = \frac{1}{1+e^{-z}}$ and β is a trainable parameter. Interesting to note that the Swish activation function is, in contrast to the ReLU, non-monotonic and with $\beta \rightarrow \infty$ becomes a standard ReLU [RZL18]. The Swish and ReLU activation functions are visualized in Fig. 2.5

2.4.3 Pooling Layer

The pooling layer is usually integrated after convolutional layers to reduce the dimensionality of a given input. In the context of CNNs, features are aggregated spatially to reduce the resolution of the feature maps and gain spatial invariant feature maps[SMB10]. Fig. 2.6 gives an example of a max pooling operation with a window size of 2 and a

stride of 2. The filter size describes the path size in x , y direction, and the stride describes the steps that are moved in between the iterations. In the max pool operation, the maximum of each block reflects the target value in the pooled image. Other pooling operations, like average pooling, are possible, but max pooling has been shown to have vastly better performance [SMB10] in vision tasks.

2.4.4 Linear Layer

The linear layer (also fully-connected or dense layer) maps the features from an input feature space to an output feature space using a weight matrix \mathbf{A} and can be expressed as

$$l(\mathbf{x}) = \mathbf{Ax} + \mathbf{b},$$

where \mathbf{b} denotes the bias and the input features \mathbf{x} are received as a flattened feature vector.

2.4.5 Training

Now that we have defined a model, we introduce the complete training routine. Note, the same model can be used with different optimization targets for different tasks like regression or classification. The focus of this thesis is on action recognition; therefore, we concentrate on presenting the methods for the classification task. Throughout the iterations over the training samples, the weights are continuously updated. For other tasks, a similar training routing is used, but with different loss functions.

Cross Entropy Loss

In a classification task, we aim to adjust the model's parameters throughout the training to infer estimated labels of given inputs that correspond to the ground-truth labels. This can be achieved by minimizing cross entropy between an input class distribution and a ground-truth class distribution. The *Cross Entropy Loss* [Cox58] is defined as:

$$\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) = - \sum_{n=1}^K \sum_{c=1}^C \mathbf{y}_n^c \cdot \log(\hat{\mathbf{y}}_n^c),$$

where \mathbf{y} is a class distribution vector for the estimated class and analog the $\hat{\mathbf{y}}$ one-hot vector for the ground-truth class [Mur12]. The loss is estimated for all class labels C over all data samples K . As we deal with multi-class problems, the cross-entropy loss in our work is defined as a categorical cross entropy loss. In practice, the loss is calculated for each batch and reduced to the overall mean.

Stochastic Gradient Descent

Gradient descent is an optimization algorithm that updates the model parameters θ with a learning rate α towards a global minimum of an objective function $J(\theta, \mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}))$, where $\mathbf{y}, \hat{\mathbf{y}}$ contain the class distributions and one-hot vectors for the complete dataset:

$$\theta_{i+1} = \theta_i - \alpha \nabla_{\theta} J(\theta, \mathcal{L}(\mathbf{y}, \hat{\mathbf{y}})),$$

where ∇_{θ} is the gradient of the objective function.

The weight initialization is crucial for the optimization using gradient descent. Various methods to initialize the parameters, e.g., drawing random weights from Gaussian distributions [KSH12] or specifically targeting ReLU activation functions by adaptive initialization methods as proposed by He et al. [He+15]. Models can also be optimized with parameters initialized by an already pre-trained model.

As the gradient descent algorithm becomes inefficient larger datasets the gradients are optimized on the sampled batches using Stochastic Gradient Descent (SGD), where the model parameters are updated for batches of a given size separately, $\mathbf{y}_{\text{Batch}}, \hat{\mathbf{y}}_{\text{Batch}}$ contain the class distributions and one-hot vectors for the batches:

$$\theta_{i+1} = \theta_i - \alpha \nabla_{\theta} J(\theta, \mathcal{L}(\mathbf{y}_{\text{Batch}}, \hat{\mathbf{y}}_{\text{Batch}})).$$

The gradients are lower per iteration, but computation becomes tractable even for larger datasets. Additional randomization of the sampled batches creates more variation of the presented training data. The parameters are optimized for several iterations over the training set, where a complete iteration over the training set is referred to as an epoch. Different strategies like training for a fixed number of epochs, early stopping when the training or validation loss is stagnating, or the validation accuracy does not increase are commonly employed.

2.5 Graph Convolutional Networks

Graph Convolutional Network (GCN)s become especially interesting over CNNs when dealing with naturally graph structured, non-euclidean, data structures like documents and their connections by citations [KW17]. Throughout this thesis, we demonstrate action recognition methods based on skeleton sequences. Skeleton sequences are also an ideal candidate for graph representations. Yan et al. [YXL18] proposed to represent skeleton sequences as a spatio-temporal graph (see Fig. 2.7). While CNNs sample pixels around the center of the filter, for a convolution on graphs the neighboring nodes \mathcal{N} are sampled, which are again partitioned and weighted individually. Given a spatial graph $G = (V, E)$, consisting of vertices $v_i \in V$ and their edges $(i, j) \in E$, where two nodes

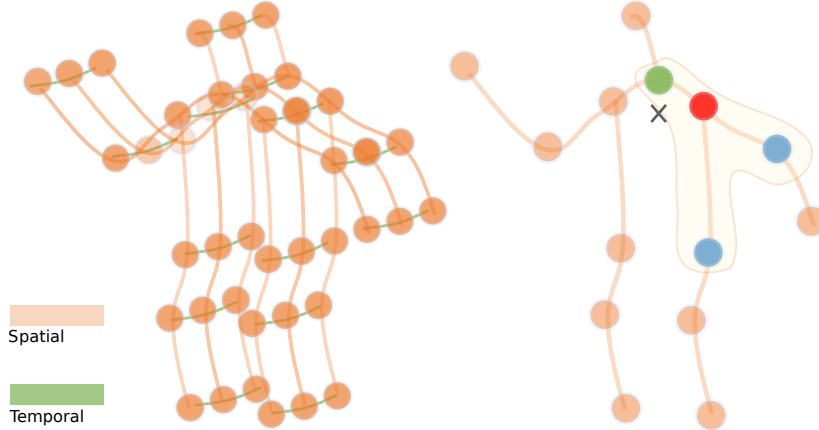


Figure 2.7: Skeleton Graph Representation for Spatio-Temporal Action Recognition. The spatio-temporal representation on the left and an partition example on the right. Figure from [Shi+19b].

are neighbors if they have a common edge. We sample the neighboring nodes of a node v_{tj} , that have a distance $d(v_1, v_2)$ below or equal to a threshold D :

$$\mathcal{N}(v_{ti}) = \{v_{tj} | d(v_{tj}, v_{ti}) \leq D\}.$$

Yan et al. [YXL18] set $D = 1$ to only consider direct neighbors in the graph.

The set of neighboring nodes is again partitioned into 3 subsets, to which vertices are associated to by the mapping function $l(v_i) : \mathcal{N}(v_i) \rightarrow \{1, 2, 3\}$. This partition is done to weight the neighbors individually by the elements contained in the subsets. $\mathcal{S}_{v_{ti}1}$ consists of the current vertex (green node in the right skeleton graph of Fig. 2.7), $\mathcal{S}_{v_{ti}2}$ consists of the vertices closer to the gravity center (centripetal subset, red node), and $\mathcal{S}_{v_{ti}3}$ contains the more distant vertices (centrifugal subset, blue nodes).

A convolution over graphs can be defined now by sampling the neighboring nodes to weight their relation to neighboring nodes with a weighting function w [YXL18]:

$$\text{gconv}(v_{ti}) = (G * w)(v_{ti}) = \sum_{v_{tj} \in \mathcal{N}(v_{tj})} \frac{1}{Z_{v_{tj}}} G_{v_{tj}} w(l(v_{tj})),$$

where the weight function is then normalized by the cardinality of the subset $Z_{v_{tj}}$ that contains the currently considered node v_{tj} .

A break-through was made by the Spatial Temporal Graph Convolutional Network (ST-GCN) [YXL18] model, that first proposed to represent skeleton sequences on a spatial and temporal level for the action recognition task on skeleton sequences. Joints and their inter-joint connections form the spatial connection, while intra-frame connections of the same joints build the temporal component. An overview of the ST-GCN

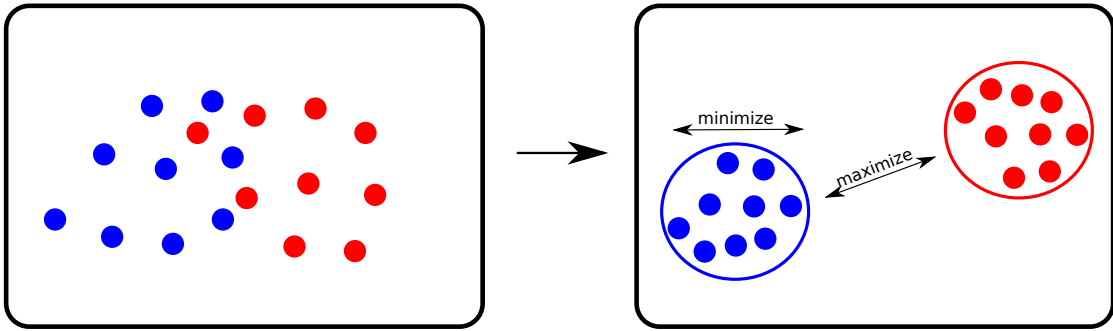


Figure 2.8: Metric Learning goal. An embedding model is trained which optimizes the projection function such that it minimizes the scatter within classes and maximizes the scatter between classes.

representation is given in Fig. 2.7. On this basis, a wide set GCN-based approaches for skeleton-based action recognition have been introduced later [Shi+19b]. For practical implementations the GCN layers are realized with an adjacency matrix representation.

2.6 Metric Learning

In contrast to a classification problem, where a class label is associated, in metric learning a projection into an embedding space on which a distance metric is defined. Similarity functions like the cosine similarity are then used to compute distances of the projected samples in the embedding space. By this, the formulation is more general in comparison to a classification formulation and allows generalizing to novel, previously unseen classes. Once a data sample or a given set of data samples are projected into a metric space, distances between samples can be used for clustering, dimensionality reduction, nearest-neighbor classification or ranking. The goal of metric learning is to learn a projection model that minimized the inter-class-scatter between sample class data samples and maximize the distance between centroids of different class sample clusters Fig. 2.8. Applications can be found in face recognition [SKP15] or person re-identification [Yi+14; WB18; HBL17].

2.7 Human Pose Estimation

Our approaches have in common that they are applicable on human pose features from sensors like RGB-D or RGB cameras. Throughout this thesis, we use mainly two kinematic models to represent the human pose (see Fig. 2.9). We therefore introduce the common approaches as used for transforming data streams from these sensors into human pose features.

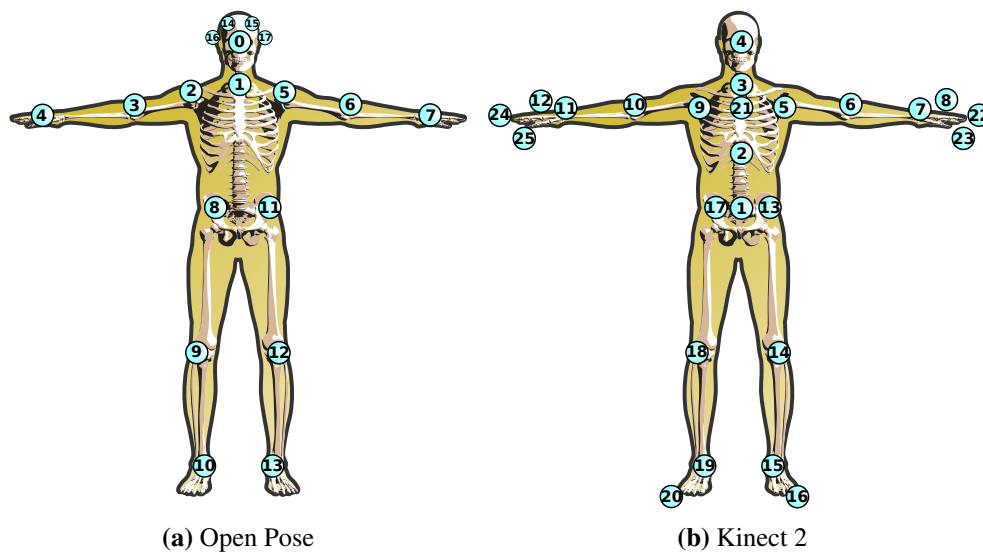


Figure 2.9: Human pose models, commonly used in human pose estimation.

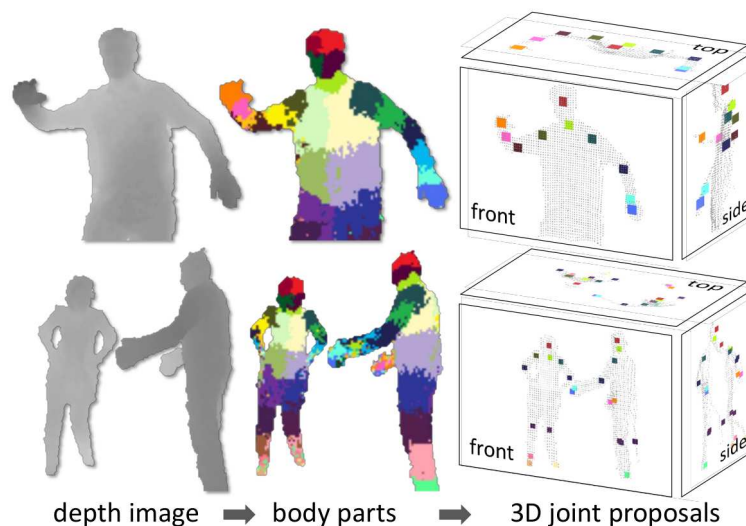


Figure 2.10: Depth-based skeleton estimation approach. Figure from [Sho+11].

2.7.1 Human Pose Estimation on Depth Images

With the release of the Microsoft Kinect [Zha12], RGB-D cameras became affordable and accessible. In contrast to common RGB cameras, an additional depth channel provides depth estimations per pixel by projecting structured light in a non-visible infrared (IR) range. With the additional IR-camera that receives the known dot light pattern and the known relative transformation between the IR-projector and the IR-camera 3D



Figure 2.11: OpenPose pose estimation approach. Figure from [Cao+21].

points can be triangulated. Many datasets like the large-scale NTU RGB+D 60 [Sha+16] and NTU RGB+D 120 [Liu+20a] datasets provide skeleton estimates that are gathered by the Kinect OpenNI SDK, which implements a human pose estimation approach by Shotton et al. [Sho+11]. The skeleton-estimation approach is depicted in Fig. 2.10. Depth images serve as the basis for the estimation of pixel regions of the body parts, which are then used to locally estimate 3D joint position estimates for multiple persons [Sho+11]. This approach is proposed for single depth images such that no temporal information is required.

2.7.2 Human Pose Estimation on RGB Images

Human pose estimation methods that operate directly on RGB images can be categorized in regression approaches and body part detection methods approaches [Zhe+20]. Recent regression methods utilize CNN that use multiple convolutions followed by linear layers to transform input images to human pose keypoints. Commonly, a L_2 distance between prediction and ground-truth pose vector is used as a loss for the training [TS14]. Body part detection methods estimate heatmaps for different body parts that are assembled to final poses [Zhe+20]. Fig. 2.11 gives an overview of the human pose estimation approach by Cao et al. [Cao+21]. This approach takes single RGB images to estimate a 2D human pose. OpenPose predicts jointly the part confidence maps and the part affinity fields. Those part affinity fields are used as a non-parametric representation to learn associated body parts from the image. In the next step, the body part candidates are matched by a bipartite matching system. And finally, all matched parts yield in pose estimates for all persons contained in the image. In this thesis, we utilized the OpenPose model [Cao+21] for some of our experiments. Recent approaches like OpenPifPaf [KBA19] utilize composite fields for a spatio-temporal association. MediaPipe by Luger et al. [Lug+19] is an highly efficient approach targeting low-latency human pose estimation among other visual estimation features. Their pose estimation approach is, however, limited to single persons, whereas OpenPose and OpenPifPaf are multi-person human-pose-estimators.

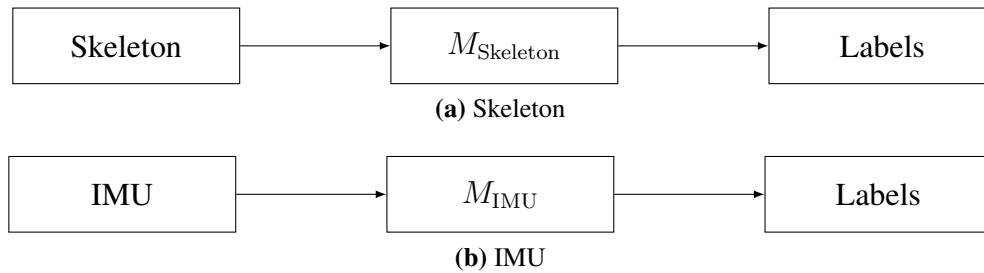


Figure 2.12: Example of uni-modal models. The architectures don't share any features or input data.

2.8 Modality Fusion

We now introduce various approaches for handling different modalities in the context of modality fusion. The recent trend in machine learning research goes into single models that generalize well across different tasks [AEP06] and modalities, i.e., by introducing the multimodal neuron [Goh+21]. In this thesis, we follow the following modality fusion definition by Luo et al. [LKL93]:

Definition 3

Multisensor fusion, ..., refers to any stage in an integration process where there is an actual combination (or fusion) of different sources of sensory information into one representational format.

Uni-Modal The *Uni-Modal* design is the most common one and widely used in image classification [KSH12; Den+09] tasks operating on single modalities. A model M is trained and inferred using a single modality in the most common case, camera images. No information fusion takes place. The resulting models solely concentrate on single modalities. An abstract example of such an approach is given in Fig. 2.12.

Multi-Modal To fuse different modalities, *Multi-Modal* architectures have been proposed. Data is either fused on a feature or representation level. In feature level fusion (also late fusion) a model is trained for each modality. Features extracted from the models are then fused to yield a final classification result. In contrast, an early fusion is a representation level fusion. An early fusion results in a single model. The benefit of early fusion methods are the reduced training cost by only training a single model. Late fusion approaches might learn more descriptive features per modality. Intermediate fusion, on a feature layer-level, has also recently been proposed [Joz+20]. Fig. 2.13 gives an abstract overview of the two different fusion concepts.

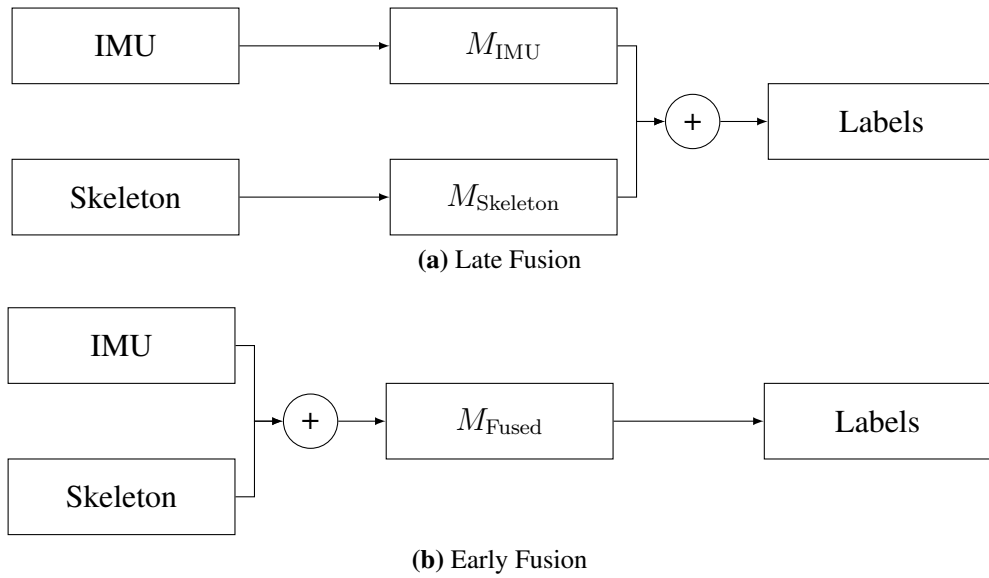


Figure 2.13: Example of multi-modal models. Modalities are fused on a feature level (late fusion) and on a representation level (early fusion).

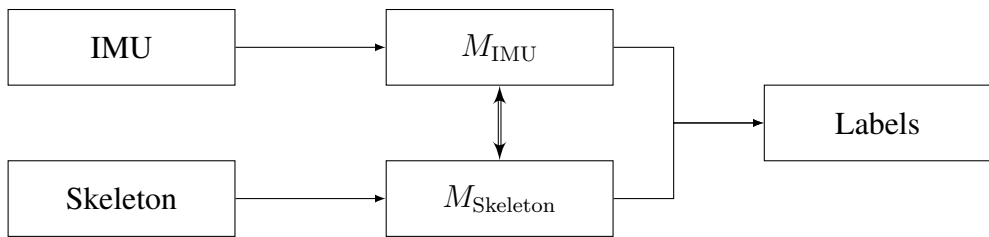


Figure 2.14: Cross-Modal feature fusion.

Cross-Modal *Cross-Modal* designs improve the models crosswise by distilling knowledge [HVD15] from models of different modalities during training or learn good candidates for common representations. During training, samples from all modalities are available. An example of such an architecture is given in Fig. 2.14. Originally, the knowledge distillation approach was proposed for model compression, where knowledge of larger deep neural networks is distilled into smaller networks to improve the model of lower complexity with information from the model of higher complexity. One can imagine distilling knowledge from a ResNet 151 teacher model into a ResNet 50 student model. One approach to distill the knowledge of a teacher model into a student model is defining the loss function as the cross entropy between the output of the student model and the output of the teacher model during training. Using knowledge distillation as a cross-modal training approach has been first proposed by Gupta et al. [GHM16] for distilling knowledge from pre-trained RGB image model to paired sample images of a

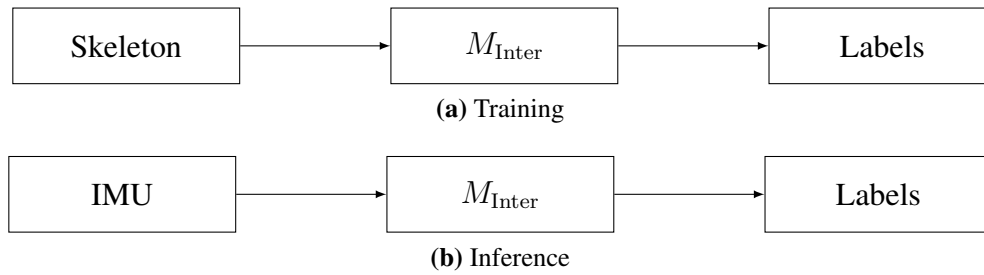


Figure 2.15: Example of an inter-modal architecture. Models that generalize well across different modalities are trained. In the given example, a model is trained using skeleton data. During inference, this model is used to infer class labels on IMU data.

different modality like depth- or optical-flow images. In the context of action recognition, these concepts have been transferred to approaches distilling knowledge, e.g., from RGB-sequences to skeleton sequences by Thoker and Gall [TG19] and across multiple sensor modalities and their combinations into a student model by Kong et al. [Kon+19].

Inter-Modal The term inter-modal originates from the freight transportation, where inter-modal containers can be used for different means of transportation like ships, trains, or trucks. The same containers can be used without adoptions to the underlying means of transportation which supports efficient transportation handling. We employ the inter-modality term in a machine learning scenario where a model is trained on a set of samples of a sensor modality that differs to the modality used during inference. In Chapter 5, we demonstrate that one-shot approaches that transform input data into a common embedding space can be used to employ an *Inter-Modal* architecture design. During training, no samples of the testing modalities are available, while the resulting model is capable of inferring results on different modalities. This protocol is challenging, avoids overfitting by hardly optimized architectures or representations, and is potentially highly practical as generalized inter-modal approaches can be transferred to novel sensor generations. Fig. 2.15 depicts an inter-modal architecture in the context of action recognition. It is to note that an inter-modal model must be designed to generalize well across different modalities. Additional augmentation that tries to transform signals from one modality into another can be either learned or designed. To the best of our knowledge, inter-modality in machine learning is not widely spread. Guo et al. [Guo+18] propose a joint inter-modal and intra-modal correlation-preservation approach to handle scenarios where only partial pairs are provided in a canonical correlation analysis, as used for dimensionality reduction or information fusion.

Chapter 3

Action Recognition on Various Sensor Data Modalities

In this and the following chapter, we present approaches for the action recognition task formulated as a supervised classification task. This chapter is based on representing the motion in images and using CNNs for their classification. The following Chapter 4 is based on GCNs. In contrast, the approaches in Chapter 5 are formulated as a semi-supervised classification task for action recognition using a single reference sample per class.

We present an approach that distinguishes from approaches in the existing literature by its focus on generalization across different sensor modalities without requiring adoptions to the underlying problem formulation, models, and training procedure. We achieve generalization by formulating the action recognition problem on a signal-level and employing a common underlying representation. A CNN is used for training the final action recognition model. We further demonstrate that this approach can fuse different modalities in an early fusion paradigm with limited training overhead, in contrast to late fusion approaches, that require separate streams per additional modality. This part of the chapter extends our prior publications for a unified action recognition approach that generalizes well across different sensor modalities [MTP20a]. We extend the underlying publication by a more detailed introduction, updated related work discussion and additional experiments on a broader set of recently published datasets. We further present action recognition results with the dense representation proposed for our one-shot action recognition approach. The wide set of experiments offers insights about the generalization capabilities and the application range that our approach is sufficient for.

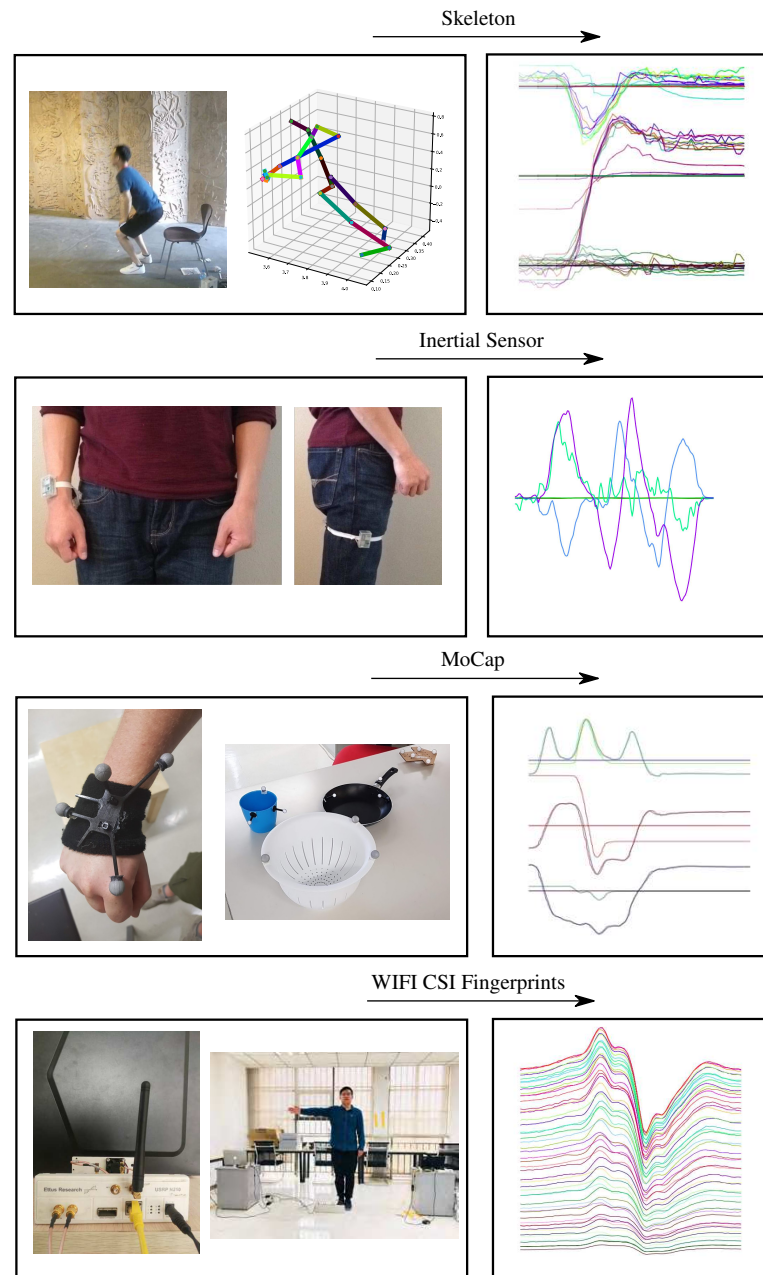


Figure 3.1: We propose a representation that is suitable for multimodal action recognition. The Figure shows representations for skeletal data from the NTU [Liu+20a; Sha+16] dataset, Inertial data from the UTD-MHAD [CJK15] dataset and Wi-Fi CSI fingerprints from the ARIL [Wan+19a] dataset.

Table 3.1: Modality support comparison of various approaches, where ✓ denotes that an approach supports a modality, ✗ denotes no support, (✓) denotes support but with additional pre-processing like extraction of human-pose features.

Name	SkI	IMU	WiFi	MoCap	RGB
Liu et al. [LLC17]	✓	✗	✗	✗	✗
Ehatisham et al. [Eha+19]	✗	✓	✗	✗	✓
Imran et al. [IR20]	✓	✓	✗	✗	✓
Liu et al. [Liu+20b]	✓	✗	✗	✗	✗
Ours	✓	✓	✓	✓	(✓)

3.1 Introduction

Action (also referred to as activity or behavior) recognition is a well-studied field and enables application in many areas like elderly care [Nou+07; ST17; NGC15; LRJ19], smart homes [NGC15; LRJ19], surveillance [Niu+04; Wil+12], driver behavior analysis [Cho+07; Mar+20; Rig+05], and robotics [Krü+07; CKG16b].

Action recognition can be defined as finding a mapping that assigns a class label to a sequence of signals. The input data can, for instance, be measurements from IMU, skeleton sequences, motion capturing sequences or image streams. In this chapter, we tackle the action recognition problem on a signal level, as this is a common basis for a variety of input modalities or features that can be transformed into multivariate signal sequences. A common basis is important for the generalization across different modalities. Table 3.1 compares an excerpt of recent approaches in terms of generalization capabilities over various modalities. Many approaches focus on tackling the action recognition problem on single modalities or multi-stream models with one stream per separate modality, which increases complexity.

Some sensors like IMUs or Wi-Fi receivers yield multivariate signals directly, other sensors like RGB-D cameras provide skeleton estimates indirectly. Skeleton estimates can be transformed easily into multivariate signals by considering their joint axes. This also holds for human poses that can be estimated on camera streams using recent methods [XWW18]. Predicting the action class from multivariate signal sequences can then be seen as finding discriminative representations for signals.

CNNs have shown great performance for image classification tasks. We, therefore, propose a representation that transforms multivariate signal sequences into images. Recent proposed CNN architectures use architecture search conditioned on maximizing the accuracy while minimizing the floating-point operations [San+18; TL19]. Therefore, they are good candidates for use in robotic systems. Figure 3.1 gives an exemplary overview of the variety of modalities that our proposed representation can be

used for. We evaluated the approach on 8 datasets containing 4 modalities. To the best of our knowledge, our signal-level action recognition approach is currently the only one supporting such a variety of modalities without special adaptations of the underlying representation or architecture design. Many proposed fusion approaches rely on custom-engineered sub-models per sensor modality, which are usually combined in multi-stream architectures. In contrast, we fuse the modalities on a representation level. This has the huge benefit of having a constant computing complexity independent of the number of modalities used, whereas multi-stream architectures raise in complexity with every modality added.

The main contributions of this chapter are as follows:

- We propose an action recognition approach based on the encoding of signals as images for classification with an efficient 2D-CNN.
- We propose filter methods on a signal level to remove signals with only a minor contribution to the action.
- We present an approach for information fusion on a signal level.

By considering the action recognition problem on a signal level, our approach generalizes well across different sensor modalities. The signal reduction prevents the image representation from overloading and allows flexible addition of signal streams. We experiment with sparse and dense representations of the underlying signals. By fusion on a signal level, we create a flexible framework for adding additional information, for instance object estimates or the fusion of different sensor modalities.

3.2 Related Work

In this section, we present action recognition methods based on traditional feature extractors and recent advances in machine learning. Existing survey papers [Zha+16; Zha+19a; WYD17; Gu+18; LKL14] do not include most recent publications, as the action recognition field is a highly active field of research. In the related work discussion, we put a focus on more recent work from the action recognition domain related to our proposed method. We put a focus on methods using skeleton sequences as input because these can be acquired on robotic systems directly from RGB-D frames or by extracting human pose features [XWW18] from video sequences. Further, large-scale benchmarks, e.g., [Liu+20a] are available for action recognition on skeleton sequences, thus a fair comparison of different approaches can be achieved.

An interesting analysis from a human visual perception perspective has been presented by Johansson [Joh73] in 1973. He found that humans are using 10-12 elements in proximal stimulus to distinguish between human motion patterns [Joh73]. This supports the use of skeletons or pose estimation maps as underlying representations for

activity recognition approaches from a visual perception perspective [Si+19]. Recent advances in action recognition developed from handcrafted feature extractors to deep learning approaches like 2D- and 3D-CNNs, while in parallel Long Short-Term Memory (LSTM) based methods also improved results on large-scale datasets. More recently, graph convolution approaches showed promising results.

Rahmani et al. [Rah+16] presented viewpoint invariant histograms of gradient descriptors for action recognition. Vemulapalli et al. [VAC14] represented skeleton joints as points in a Lie-group. The classification is then done by a combination of dynamic time-warping [BC94], Fourier temporal pyramid representation and linear Support Vector Machine (SVM) [VAC14]. More recent approaches suggest representing skeleton sequences as images and 2D-CNNs for recognition. Wang et al. [Wan+18a] encode joint trajectory maps into images based on three spatial perspectives. Caetano et al. [Cae+19; CBS19] represent a combination of reference joints and a tree-structured skeleton in images. Their approach preserves spatio-temporal relations and joint relevance. Liu and Yuan [LY18] study a pose map representation. The approach that comes closest to our approach is by Liu et al. [LLC17]. Liu et al. presented a combination of skeleton visualization methods and jointly trained them on multiple streams. In contrast to our approach, their underlying representation enforces custom network architectures and is constrained to skeleton sequences, whereas our approach adds flexibility to other sensor modalities. Kim et al. [KR17] presented a visual interpretable method for action recognition using temporal convolutional networks. Their approach uses a spatio-temporal representation, which allows visual analysis to understand why a model predicted an action. Especially joint contributions are visually interpretable.

3D CNN 3D convolutions for video action recognition was popularized by Tran et al. [Tra+15]. They have shown good performance on direct video action classification. A three-stream network has then been proposed to integrate multiple cues sequentially via a Markov chain model [Zol+17]. By the integration of additional cues from e.g., pose information, optical flow and RGB images using a Markov chain they could increase the recognition accuracy incremental with each additional cue. A special focus on more complex actions was put by Hussein Hussein et al. [HGS19]. They use multiscale temporal convolutions and thereby reduce the complexity of 3D-CNN architectures to show increased performance on more complex, longer activities. With the SlowFast [Fei+19] proposed a two pathways 3D-CNN. The slow path, at a low frame rate, aims at capturing spatial semantics, while the fast path, with a high frame rate, focuses on motion at a fine temporal resolution. Their two path approach has shown great performance in various video classification tasks. Carreira and Zisserman [CZ17] presented an Inflated 3D-CNN. They propose to initialize a 3D-CNN with pre-trained 2D-CNN weights, e.g., from ImageNet [KSH12] pre-training. Filters and pooling kernels of deep networks can then be expanded to 3D while leveraging from already established 2D-CNN archi-

tures and their parameters. They proposed single stream networks, that operate on RGB sequences and optical flow individually, but can also be fused with two streams, improving the model's performance.

Time Series Classification Commonly, for time series classification, time series are transformed to feature vectors with a sliding window approach and then are analyzed with a machine learning approach [SL17]. With WEASEL by Schäfer and Leser [SL17], presented a time series classification approach that has a low amount of very discriminative features by using a bag-of-pattern approach. Hochreiter and Schmidhuber [HS97] proposed LSTM tackle the back-flow issues of Recurrent Neural Networks (RNNs) by incorporating memory cells and gated connections. More recently, approaches based on transformer networks [Vas+17] have shown great performance for time series classification [Zer+21; OWW18]. CNN architectures for signal classification have also been studied previously in audio processing [Her+17]. ResNet 1D-CNN architectures have been used for joint classification and localization of activities in Wi-Fi -signals [Wan+19a]. For activity classification on a set of inertial sensors Yang et al. [Yan+15] acquire time-series signals and classify the activities using a multi-layer CNN.

Skeleton-based Action Recognition A good indicator for the progress of skeleton-based action recognition are the results on the NTU RGB+D dataset [Liu+20a]. Initially, approaches have been based around LSTM [Liu+16a] or RNNs. For skeleton-based action recognition, approaches based on GCN are defining current state-of-the art methods. With the spatio-temporal GCN approach by Yang et al. [Yan+19] steadily improved action recognition on skeleton sequences [Liu+20b; Pap+20]. Liu et al. [Liu+16a] presented a spatio-temporal LSTM inspired by graph-based representation of the human skeleton. They further introduced a novel trust-gating mechanism to overcome noise and occlusion. Si et al. [Si+19] presented an Attention Enhanced Graph Convolutional LSTM Network (AGC-LSTM). They use feature augmentation and a three-layer AGC-LSTM to model discriminative spatial-temporal features and yield good results on cross-view and cross-subject experiments on skeleton sequences. Very recently Papadopoulos et al. [Pap+20] proposed two novel modules to improve action recognition based on Spatial Graph Convolutional [YXL18] networks. The Graph Vertex Feature Encoder learns vertex features by encoding skeleton data into a new feature space, while the Dilated Hierarchical Temporal Convolutional Network introduces new convolutional layers capturing temporal dependencies. Very recently, Song et al. [Son+21] introduced the concept of compound scaling from EfficientNet CNN [TL19] to GCNs and demonstrated state-of-the-art performance on the NTU RGB+D 120 dataset while reducing the number of required parameters. Note, all the here mentioned approaches focus on only skeleton-based action recognition, whereas our approach generalized across different modalities as well.

Multi-modal Action Recognition Most fusion methods rely on complex individual representations per modality or propose complex multi-stream architectures. In contrast, our approach allows modality fusion using matrix concatenations in a single stream. By this, our approach is directly usable for a variety of sensors used in robotics like inertial measurement units, Motion Capturing System or skeleton sequences and can integrate features extracted from higher dimensional image streams that result, e.g., in human pose features [XWW18]. Interesting fusion approaches have been presented previously. Perez-Rua et al. [Per+19] presented an approach for multi-modal fusion architecture-search using RGB, depth and skeleton fusion. Song et al. [Son+18] extract visual features from different modalities around skeletal joints from RGB and optical flow representations. Whereas those approaches have focused on multiple modalities originating from one device (e.g., Microsoft Kinect) there are also methods for the fusion of sensor data from different devices. Imran and Raman [IR20] propose a three-stream architecture, with different sub-architectures per modality. A 1D-CNN for gyroscopic data, a 2D-CNN for a flow-based image classification and an RNN for skeletal classification. In the end, individual features are fused, and a class label is predicted. The fused results are promising, and additional modalities improved the results. Additional augmentation by signal filter methods has shown to influence the result positively as well. However, the complexity of the architecture and their sub-architectures require engineering and training overhead and lead to increased run-times by each added modality. This is an issue that we overcome by using a common representation for various modalities. Chen et al. [CJK15] fuse depth information, inertial and demonstrate positive influence. However, they also use two different approaches for each modality. Namely, they use depth motion maps for depth sequences and partitioned temporal windows for signal classification of the gyroscope signals.

3.3 Approach

The problem of action recognition with a given set of k actions $Y = \{0, \dots, k\}$ can be reformulated as a classification problem, where a mapping $f : \mathbb{R}^{N \times M} \rightarrow Y$ with N being the number of signals and M relating to the measurement samples M must be found that assigns an action label to a given input. The input in our case is a Matrix $\mathbf{S} \in \mathbb{R}^{N \times M}$ where each row vector represents a discrete 1-dimensional signal and each column vector represents a sample of all sensors at one specific time step.

After signal reduction, the reduced signal matrix $\mathbf{S}_{\text{focused}}$ is transformed to an RGB image $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$ by normalizing the signal length M to W and the range of the signals N to H . The identity of each signal is encoded in the color channel. An overview of our approach is given in Fig. 3.2.

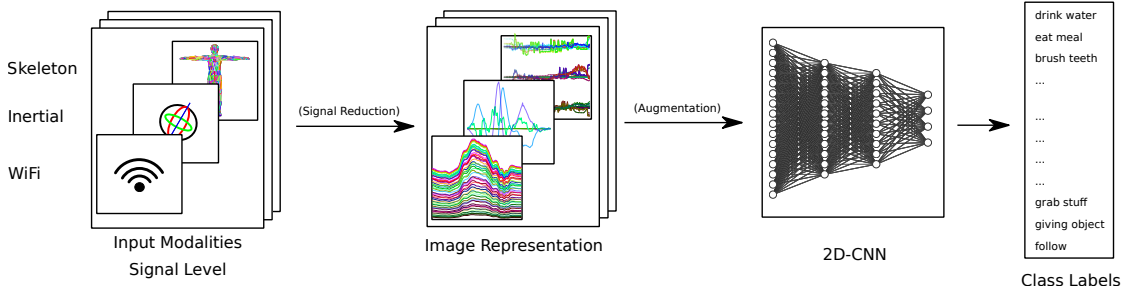


Figure 3.2: Approach overview. We propose to transform individual signals of different sensor modalities and represent them as an image. The resulting images are then recognized using a 2D convolutional neural network.

3.3.1 Signal Reduction

To avoid cluttering of the signal representation, we propose a straightforward method for signal reduction which can be used across different modalities. This allows to lay focus on signals with high information content while removing the ones with low information content.

If for example sequences of skeletons are considered, many of the joints are not moving significantly throughout the performance of an action. Intuitively, it can be understood that when an action is performed while standing in one place, the signal of the leg movement does not contribute much to help in classifying the performed action. From this intuition, we developed the assumption that low variance signals do contain less information in the context of action recognition as high variance signals. Therefore, we propose to set the signals to zero which are not actively contributing to the action by applying a threshold τ to the signal's standard deviation σ . In our experiments τ was defined as 20% of the maximum value of all signals.

To be more concise, we define the decision function $\text{filter}(\mathbf{s}_j)$ for the j -th signal \mathbf{s}_j in matrix \mathbf{S} as

$$\text{filter}(\mathbf{s}_j) = \begin{cases} 1, & \text{if } \sigma(\mathbf{s}_j) \geq \tau \\ 0, & \text{otherwise.} \end{cases}$$

When applying this function to each signal in matrix \mathbf{S} we receive a vector $\mathbf{c} \in \mathbb{R}^N$ which encodes in each element if the corresponding signal contributes to the action. By element-wise multiplication of each column vector of \mathbf{S} with \mathbf{c} $\mathbf{S}_{\text{focus}}$ is received where all signals that do not contribute to the action are set to zero. The signals with low contribution to actions are not removed but set to zero to prevent losing the joint identity (encoded in different colors).

Reducing the signals with low contribution to the action reduces the amount of overlapping signals in the image representation, which in turn allows increasing the total number of fused signals. We suggest applying signal reduction before fusion because

different scaling of sensor data can result in the elimination of all signals of a sensor with lower variance as another.

3.3.2 Signal Fusion

By our formulation, the fusion of signals becomes a matrix concatenation:

$$\mathbf{S}_{\text{fused}} = (\mathbf{S}_1 | \mathbf{S}_2),$$

where $\mathbf{S}_{\text{fused}}$ is the fusion of \mathbf{S}_1 and \mathbf{S}_2 under the assumption that both matrices have the same number of columns, where columns represent the sequence length. This can be either achieved by subsampling the higher frequency signals or interpolating the lower frequency signals. An example of sensor fusion is the encoding of multiple identities from skeletal data with $\mathbf{S}_{\text{fused}} = (\mathbf{S}_{\text{id1}} | \mathbf{S}_{\text{id2}})$, where two identities are fused. Another example is fusion of two sensor modalities with, i.e., $\mathbf{S}_{\text{fused}} = (\mathbf{S}_{\text{skeleton}} | \mathbf{S}_{\text{inertial}})$ or adding interaction context by $\mathbf{S}_{\text{fused}} = (\mathbf{S}_{\text{skeleton}} | \mathbf{S}_{\text{objects}})$. We therefore created a simple framework to support a wide variety of possible applications.

3.3.3 Sparse Representation

To allow a CNN based classifier to discriminate well between the action classes, we aim to find a discriminative representation in the first place. For encoding the signal identity, we sample discriminative colors in the HSV color space depending on the number of signals. Similar to the sparse representation proposed by Liu et al. [LLC17], we make the initial assumption that temporal relations are represented by the position in the image. However, network architectures of lower depth seem not to maintain a global overview of the input but focuses on local relations. Therefore, we encode local temporal information by interpolating from white to the sampled color throughout the sequence length. Signal changes are encoded spatially and joint relation are preserved. Fig. 3.3 and Fig. 3.4 give exemplary representations for skeleton and inertial sequences (Fig. 3.3) and Wi-Fi CSI fingerprints (Fig. 3.4). A limitation of this approach is that only lower dimensional signals can be encoded. Image sequences or their transformations like optical flow motion history images are too high dimensional to encode on a signal level by using our representation. Extracted human pose estimates, hand- or object estimates from image sequences are adequate signals for encoding in this representation.

3.3.4 Dense Representation

We also experimented with a dense representation, as proposed in [MTP20b] for one-shot action processing. Multivariate signal or higher-level feature sequences are re-assembled into a 3 channel image. Each row of the resulting image corresponds to one

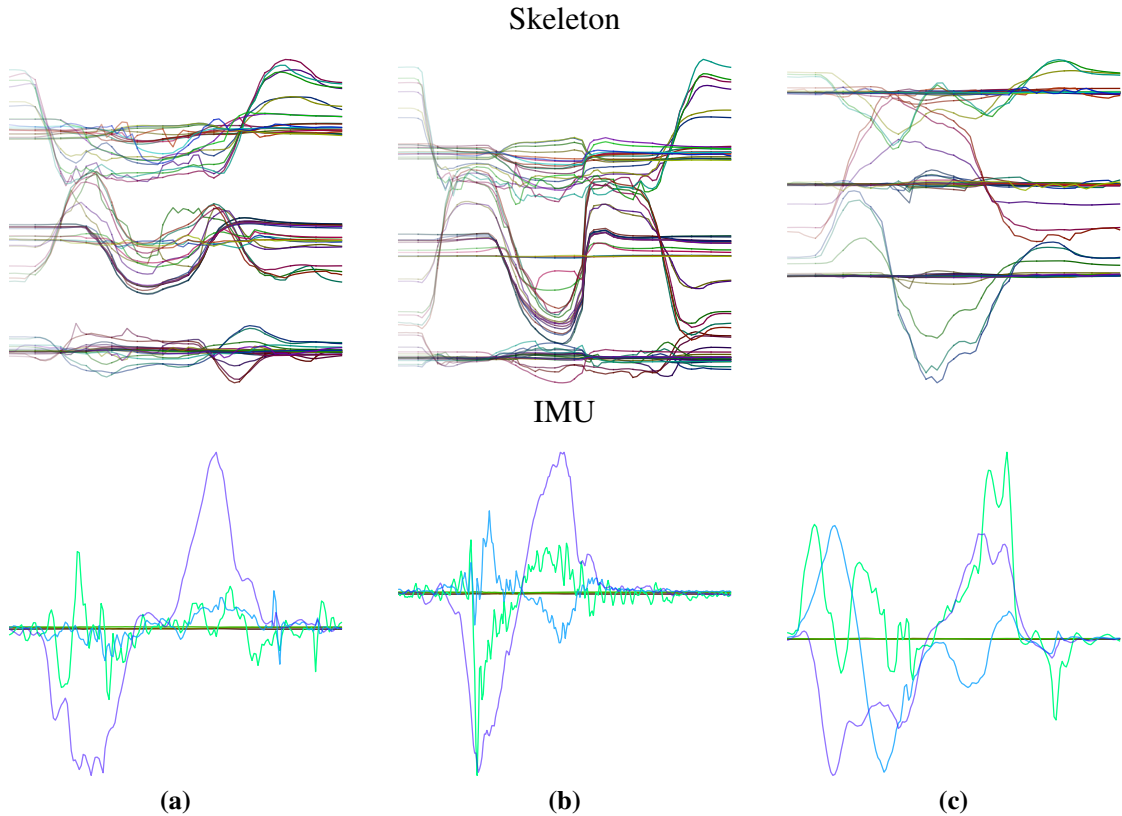


Figure 3.3: Sample representations of the UTD-MHAD dataset: (a) and (b) represent the same class (a27) of different subjects. (c) is a sample of a different class (a1). The color encoded lines correspond to the joint signals. On the top the representation for skeletal data is shown and on the bottom their respective inertial data. Note: the axes are intentionally omitted, as these are depicted as the actual underlying representations.

joint, and each channel corresponds to one sample in the sequence. The color channels, red, green and blue, represent respectively the signals' x-, y- and z-values. The resulting images are normalized to the range of 0 to 1. We chose to normalize over the whole image to preserve the relative magnitude of the signals. We experimented with a joint-level normalization, which can be interpreted as a per-row normalization but found the results to be negatively affected as spatio-temporal inter-joint relation get lost with this normalization. In contrast to the previously presented sparse representation for various sensor modalities [MTP20a] or skeleton-based action recognition [Wan+18a; LLC17] the proposed representation is invertible and more compact. Example representations are shown in Fig. 3.5.

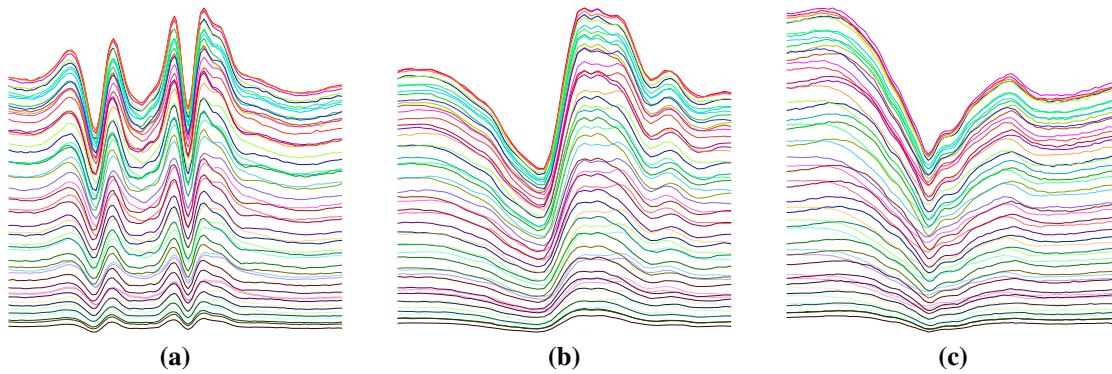


Figure 3.4: Sample representations from the Wi-Fi CSI fingerprints of the ARIL dataset [Wan+19a]. (a) and (b) represent the same class (0) of different subjects. (c) represents a different class.

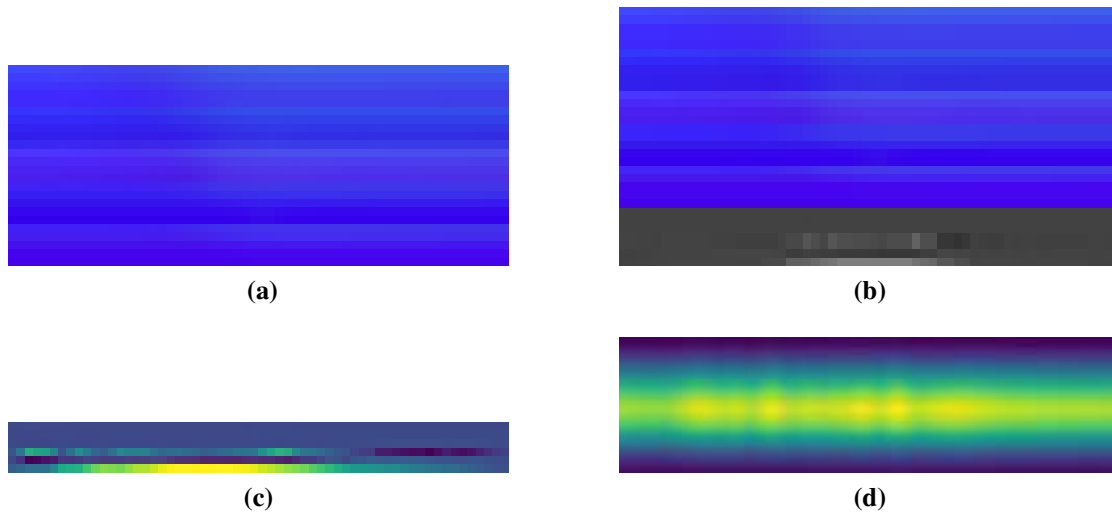


Figure 3.5: Example representations for skeleton sequences (a), inertial measurements (c), fused measurements (b) with skeleton and sub-sampled IMU measurements, and Wi-Fi CSI fingerprints (d). The four example representations show the range of modalities we conducted experiments on. Axes are intentionally omitted. Joints are represented in the y-axis, temporal information is encoded in the x-axis.

3.3.5 Augmentation

Augmentation methods have shown to influence the generalization successfully. In our case, we can create artificial training data on a signal level by interpolating, sampling, scaling, filtering, adding noise to the individual signals or augment the resulting image

representation. Liu et al. [LLC17] already proposed to synthesize view independent representations for skeletal motion. As we consider action recognition on a signal level, these transformations would result in augmentations integrated as a pre-processing step for each modality separately. Therefore, we decided to focus on augmenting the resulting image representation, which can be efficiently integrated into training pipelines. Augmentation applied to the image representation during training still allows interpretation of an effect on the underlying signals. Stretching the width describes the same action but executed in a different speed, while perspective changes or rotations can synthesize slightly different executions during the demonstrations.

3.3.6 Architecture

Most action recognition approaches based on CNNs present custom architecture designs in their pipelines [LLC17; Ke+17]. A benefit is the direct control over the number of model parameters and can be specifically engineered for data representations or use cases. Recent advances in architecture design cannot be transferred directly. Searching good hyperparameters for training is then often an empirical study. Minor architecture changes can result in an entirely different set of hyperparameters. He et al. [He+16] suggested the use of residual layers during training, resulting in more stable training. Tan et al. [TL19] recently proposed a novel architecture category based on compound scaling across all dimensions of a CNN. We take advantage of the recent development in architecture design and use an already established architecture for image classification. The recently proposed EfficientNet [TL19] architecture is of special interest in the robotics context, as it's based on an architecture search, conditioned on maximizing the accuracy while minimizing the floating-point operations.

3.3.7 Implementation

Our implementation is done in PyTorch Lightning [Pas+19; Fal19], which puts a focus on reproducible research. Hyperparameters and optimizer states are logged directly into the model checkpoints. The source code is made publicly available. We used a re-implementation and pre-trained weights of the EfficientNet [TL19] architecture. For training, we used a Stochastic Gradient Decent optimizer with a learning rate of 0.1 and reduction of learning rate by a factor of 0.1 every 30 epochs with a momentum of 0.9. The learning rate reduction was inspired by He et al. [He+16]. A batch size of 40 was used on a single Nvidia GeForce RTX 2080 TI with 11 GB GDDR-6 memory. We trained for a minimum of 150 epochs and used an early stopping policy based on the accuracy after. Similar model checkpoints were created for an increased validation accuracy. For optimizing the training, we used a mixed precision approach by training using 16bit float with a 32bit float batch-norm and master weights. A gradient clipping of 0.5 prevented gradient and loss overflows.

3.4 Experiments

In this section, we present the datasets that we used in our evaluation of our action recognition approaches. Further, we give an overview of the most common evaluation protocols. We conducted experiments on 8 different datasets. The NTU RGB+D 120 [Liu+20a], UTD-MHAD [CJK15], ARIL [Wan+19a] and the Simitate [Mem+19a] dataset. These datasets contain in total 4 modalities. In addition to the original publication we experimented on other datasets that demonstrate generalization capabilities across various applications, setups and number of classes. Those datasets are the ETRI-Activity-3D [Jan+20], Toyota Smarthome [Das+19], UAV-Human [Li+21] and the Kinetics 400 [CZ17] datasets. Skeleton sequences are evaluated on the recently released NTU RGB+D 120 [Liu+20a] and the UTD-MHAD dataset [CJK15]. The NTU RGB+D 120 dataset demonstrates the scaling capabilities of our approach as it contains 120 classes in more than 114000 sequences. The UTD-MHAD dataset [CJK15] provides 27 classes but includes IMU data beside the skeleton estimates. Therefore, it is suitable to demonstrate the cross modal capabilities of our approach. We further use it for our fusion experiments. We further executed experiments on activity recognition datasets containing Wi-Fi CSI fingerprints [Wan+19a] and Motion Capturing data from the Simitate [Mem+19a] dataset. The ETRI-Activity-3D and Toyota Smarthome dataset have a focus on assisted living for elder people in a smarthome. The UAV-Human dataset shows applicability from an UAV perspective and the Kinetics 400 dataset is used for highly varying videos sourced from YouTube and further contains 400 different action classes. For our experiments, we generated the representations of the datasets prior and used an EfficientNet-B2 [TL19] architecture for classification. AIS in the tables denotes the additional augmentation of the training signals in image space. Results are compared to other approaches in the Results Section 3.4.3.

3.4.1 Evaluation Protocols

Common proposed protocols for the evaluation of action recognition approaches are the cross-setup (see Fig. 3.6) and cross-subject (see Fig. 3.7) protocols. The cross-setup protocol aims at benchmarking the generalization capabilities across different setups in terms of varying camera or subject positions, varying backgrounds and locations. In the NTU RGB+D 120 dataset for instance are 32 different setups are used to build the dataset [Liu+20a]. In this thesis, we denote cross-view protocols as cross-setup protocols, but state at the relevant positions also the protocol name from the original dataset. Cross-setup protocols are contained in the NTU RGB+D 60 [Sha+16] and NTU RGB+D 120 [Liu+20a], the MMAAct [Kon+19], the Toyota Smarthome dataset [Das+19]. Examples for cross-setup protocols are given in Fig. 3.6.

Commonly, half of the setups is used for training and the remainder for testing. The cross-subject evaluation protocol in contrast aims at testing the generalization capabil-

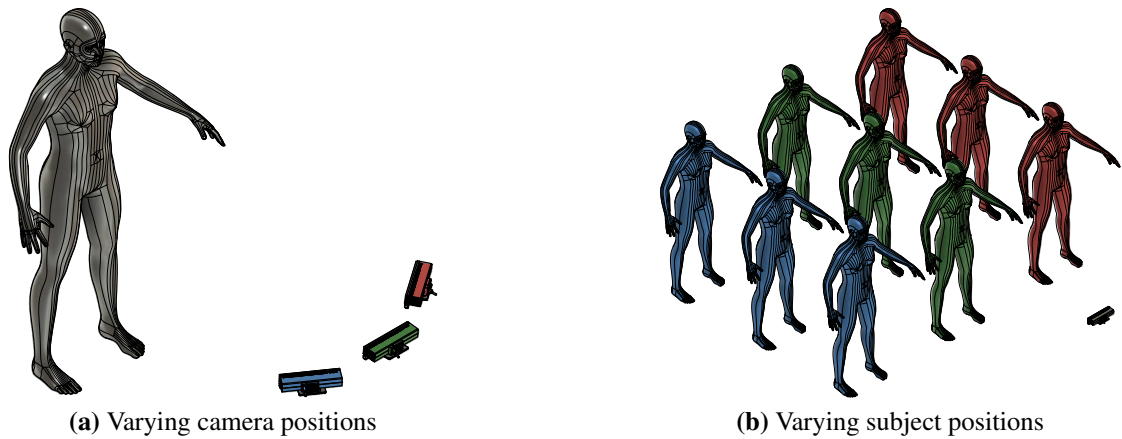


Figure 3.6: An evaluation following the cross setup protocol. An example is given in (a), e.g., when trained on the green camera and tested on the blue and red cameras. The cross setup also translates to subject locations (b), e.g., when trained on positions for green and tested on blue and red.

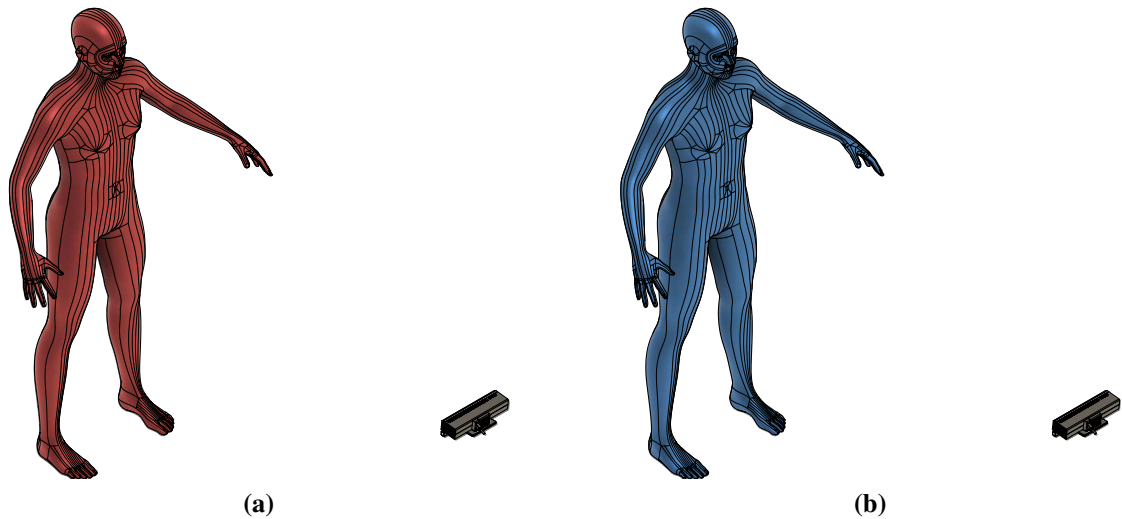


Figure 3.7: An evaluation following the cross subject protocol trains on a set of subjects and tests on a set of unseen subjects. In this example, the colors in (a) and (b) denote different subjects.

ities of a model in terms of varying subjects. Datasets containing a cross-subject protocol are the UTD-MHAD [CJK15], the NTU RGB+D 60 [Sha+16] and NTU RGB+D 120 [Liu+20a], the ETRI dataset [Jan+20], the Toyota Smarthome dataset [Das+19], the UAV-Human [Li+21]. Examples for cross-subject protocols are given in Fig. 3.7.

3.4.2 Datasets

Table 3.2: Action recognition datasets used in the experiments. Note: * denotes datasets on which experiments were conducted after the initial publication, ¹ denotes that these datasets were used for our CNN based action recognition and ² for our GCN based action recognition approaches. For the Kinetics 400 dataset, where denoted with ³ we use the pose-features as pre-calculated in [YXL18].

Dataset	Splits	Modalities	Classes	Subjects	Setups	Samples
UTD-MHAD ^{1,2} [CJK15]	Subject	RGB+D, Skl, IMU	27	8	1	861
NTU RGB+D 120 ¹ [Liu+20a]	Subject, Setup	RGB+D, Skl, IR	120	106	32	114,480
Toyota Smarthome ^{*,1} [Das+19]	Subject, Setup	RGB, Skl	31	18	8	16,129
ETRI Activity ^{*,1} [Jan+20]	Subject	RGB+D, Skl	55	100	-	112,620
UAV-Human ^{*,1} [Li+21]	Subject	RGB, Skl	155	119	-	67,428
ARIL ¹ [Wan+19a]	80/20	Wi-Fi	6	-	16	1,394
Simitate ¹ [Mem+19a]	80/20	RGB+D, MoCap	27	8	-	1,938
Kinetics 400 ^{*,1} [CZ17; YXL18]	92/8 ³	RGB, Skl ³	400	-	-	260.000 ³

In the following, the datasets on which the experiments were performed are introduced. We aim at showing generalization capabilities of our approach across different sensor modalities and various applications. Therefore we experimented on 8 containing 4 different modalities. We use the datasets listed in Table 3.2 in our experiments.

NTU RGB+D 60 / NTU RGB+D 120

The NTU RGB+D 120 [Liu+20a] dataset is a large-scale action recognition dataset containing RGB+D image streams and skeleton estimates. In contrast to the first NTU RGB+D 60 version of the dataset which contained 56880 sequences with 60 classes, the extended NTU RGB+D 120 dataset consists of 114,480 sequences containing 120 action classes from 106 subjects in 155 different views. Cross-view and cross-subject splits are defined as protocols. For the cross-subject evaluation, the dataset is split into 53 training subjects and 53 testing subjects, as reported by the dataset authors [Liu+20a]. For the cross-setup evaluation, the dataset sequences with odd setup IDs are reserved, while the remainder is used for training. Resulting in 16 setups used during training and 16 used for testing. We report results on both versions with both cross subject and cross view splits.

UTD-MHAD

This dataset [CJK15] contains 27 actions of 8 individuals performing 4 repetitions each. RGB-D camera, skeleton estimates and inertial measurements are included. The RGB-D camera is placed frontal to the demonstrating person. The IMU is either attached

at the wrist or the leg during the movements. A cross-subject protocol is followed as proposed by the authors [CJK15]. Half of the subjects are used for training, while the other half is used for validation. This dataset is a great candidate because it contains various data modalities and also allows fusion experiments. Because of its different modalities, we use it for experiments on skeleton, inertial and fused data.

ARIL

This dataset [Wan+19a] contains Wi-Fi Channel State Information (CSI) fingerprints. The CSI describes how wireless signals propagate from the transmitter to the receiver. A standard IEEE 802.11n Wi-Fi protocol was used to collect 1398 CSI fingerprints for 6 activities. The data is varying by location. The 6 classes represent hand gestures *hand circle*, *hand up*, *hand cross*, *hand left*, *hand down*, and *hand right* targeting the control of smart home devices. For our experiments, we use the same train/test split as was used by the authors of the dataset (1116 train sequences / 278 test sequences).

Simitate

The Simitate [Mem+19a] benchmark focuses on robotic imitation learning tasks. Hand and object data are provided from a motion capturing system in 1932 sequences containing 27 classes of different complexity. The individuals execute tasks of different kinds of activities, from drawing motions with their hand-over to object interactions and more complex activities like ironing. This dataset is interesting as we can fuse human and object measurements from the motion capturing system to add context information. Good action recognition capabilities will allow direct application to symbolic imitation approaches. We use an 80/20 train/test split for our experiments.

ETRI

The ETRI-Activity3D [Jan+20] dataset consists of 112,620 samples containing 55 activity classes recorded from 100 subjects. The activities were chosen visiting elderly people above the age of 70, in 53 homes, observing the most frequent activities. The dataset was then constructed from two age groups. The first group consists of 50 elderly people aged between 64 aged 88 years in order to gather realistic inter-class variation of the actions. The second group consists of 50 younger subjects in their 20s.

Toyota Smarthome

The Toyota smarthome [Das+19] dataset provides more than 16.000 RGB-D sequences separated into 31 action classes performed by 18 elderly people in a smarthome. In contrast to datasets like, e.g., NTU or the UTD-MHAD dataset, the Toyota smarthome dataset is completely unscripted. In total, 3 different scenes observed with 7 cameras

are provided. This dataset has practical classes like *take pills* to assist elder people in smartly in their daily live. Cross-subject and cross-view protocols are proposed as protocols.

UAV-Human

The UAV-Human benchmark [Li+21] aims at various challenges in understanding human behavior from unmanned aerial vehicles. In this work, we concentrate on the action recognition task, which contains 67,428 multi-modal video sequences with 119 subjects. In total, 115 different action classes are divided into 6 categories (daily activities, productive activities, violent activities, social interaction behaviors, life-saving activities and UAV control gestures). The UAV was equipped with a night vision, a fish-eye and an Azure Kinect DK RGB-D camera. The provided skeleton-data originates from the Region Multi-person Pose Estimator (RMPE) [Fan+17b]. Two cross-subject based evaluation protocols are proposed (CSv1, CSv2). They differ in the subject IDs used for training and testing. Both protocols use 89 subjects for training and 30 for testing.

Kinetics 400

The Kinetics 400 [CZ17; Kay+17] dataset is an interesting dataset for large-scale evaluation. The dataset consists of 400 classes, with at least 400 samples per class from YouTube videos. Videos were obtained by matching video titles and action list and then verified by a manual label process. The dataset is especially interesting because of the unconstrained setup and the large scale. However, the videos can disappear over time, leading to a changing dataset over time. In our work, however, we use the pre-calculated pose features from Yan et al. [YXL18] for 260.000 sequences that remain available even if the original videos might disappear.

3.4.3 Results

We did our best to include results from the most recent approaches for comparison. We found that the proposed representation on a signal level archived good performances across different modalities. An improvement of +6.8% over the baseline has been achieved on a Wi-Fi CSI fingerprint-based dataset [Wan+19a]. Augmentation has shown a positive impact on the resulting accuracy across modalities. The resulting model based on an EfficientNet-B2 performs well in interpreting spatial relations on the color encoded signals across the experiments. Results denoted with * have been added after the initial publication of our approach.

NTU RGB+D 120 For the NTU RGB+D 120 dataset, we give results in Table 3.3. Related results are taken from literature [Liu+20a; Cae+19; Pap+20; Che+20; Son+21].

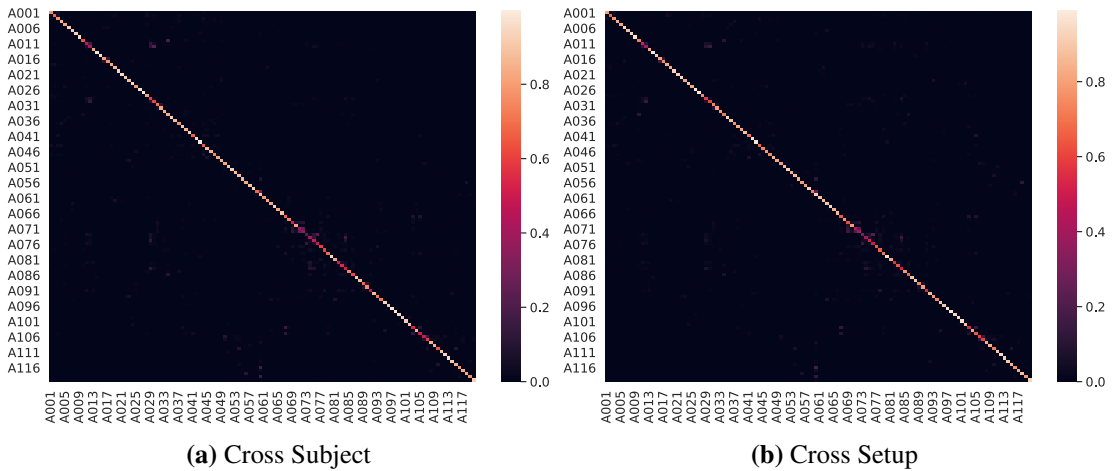


Figure 3.8: Confusion matrices for the NTU RGB+D 120 dataset for both, the cross setup and cross subject, splits.

A skeleton with 25 joints serves as input for the training of our model. In case multiple identities are contained, they are fused with the presented signal fusion approach. We got a cross-subject accuracy of 70.8% and a cross-view accuracy of 71.6% for our sparse representation. For the dense representation, we yield an accuracy of 80.0% for the cross-subject protocol and 82.0% for the cross-view split. For both dense models, results without investment of dataset-specific model tuning are reported. Confusion matrices for both splits, using the dense representation without additional augmentation, are shown in Fig. 3.8. Even so, the splits are quite different, the resulting models face issues with similar classes. Intuitively, when considering sequential data, LSTM based approaches are considered. We highly outperform the LSTM based approaches [Sha+16; Liu+16a; Liu+17b; Liu+18b]. More directly comparable are CNN based approaches [LLC17; Ke+18; LY18; CBS19; Cae+19]. All the mentioned approaches concentrate on finding representations limited to skeleton or human pose features, while our approach considers action recognition on a signal level and therefore is transferable to other modalities as well. The discriminative representation we suggest comes closest to the one by Liu et al. [LLC17]. With the proposed augmentation method and the EfficientNet-B2 based architecture, we outperform the current CNN based approaches by +2.9% (cross-subject), +4.6% (cross-view) with the sparse representation. Using the dense representation, we improve over the TSRJI CNN baseline by +12.1% (cross-subject) and +19.2% (cross view). In that case, our approach performs better on the cross view protocol in contrast to some related CNN-based approaches (TSRJI, SkeleMotion, Multi-Task CNN) approaches that show higher accuracies on the cross-subject split than on the cross-view split. Papadopoulos et al. [Pap+20] presented an approach based on a graph convolutional network that performs 5.7% better on the cross-subject

Table 3.3: Results on NTU RGB+D 120. Units are in %.

Approach	CS	CV
Part Aware LSTM [Sha+16]	25.5	26.3
Soft RNN [Hu+19]	36.3	44.9
Spatio-Termoral LSTM [Liu+16a]	55.7	57.9
GCA-LSTM et al. [Liu+17b]	58.3	59.2
Skeleton Visualization (Single Stream) [LLC17]	60.3	63.2
Two-Stream Attention LSTM [Liu+18b]	61.2	63.3
Multi-Task CNN with RotClips [Ke+18]	62.2	61.8
Body Pose Evolution Map [LY18]	64.6	66.9
SkeleMotion [Cae+19]	67.7	66.9
TSRJI [CBS19]	67.9	62.8
<i>Ours (Sparse, AIS)</i>	70.8	71.6
ST-GCN + AS-GCN w/DH-TCN [Pap+20]	78.3	79.2
<i>Ours (Dense)*</i>	80.0	82.0
4s Shift-GCN [Che+20]*	85.9	87.6
Efficient-GCN-B4[Son+21]*	88.7	89.1

split and 8% better on a cross-view split than our approach with the sparse representation. With the dense representation, we perform better than the ST-GCN-based approach by Papadopoulos et al. [Pap+20]. Recent advances in GCN-based approaches for skeleton-based action recognition outperform our approach by a large margin, like the 4s Shift-GCN [Che+20] and the Efficient-GCN-B4 [Son+21]. All the approaches we compared to focus on the recognition for a single modality, whereas our approach shows good results while generalizing well to other modalities.

UTD-MHAD Results on the UTD-MHAD dataset are shown in Table 3.4. We compare our approach to the baseline of the authors as well as a more recent approach [Zha+19b; Wan+18a]. While Zhao et al. [Zha+19b] perform better than our proposed approach, we get slightly better results than Wang et al. [Wan+18a] and further have the benefit of being applicable on other sensor modalities. It is to note that the perfect accuracy of 100.0% in [Zha+19a] was falsely reported on a similar named dataset. Fused experiments are executed by fusing skeleton estimates and inertial measurements $\mathbf{S}_{\text{fused}} = (\mathbf{S}_{\text{skeleton}} | \mathbf{S}_{\text{inertial}})$. We improve the UTD-MHAD inertial baseline [CJK15] by +14.4% using the sparse representation with augmentation and an additional +3.3%

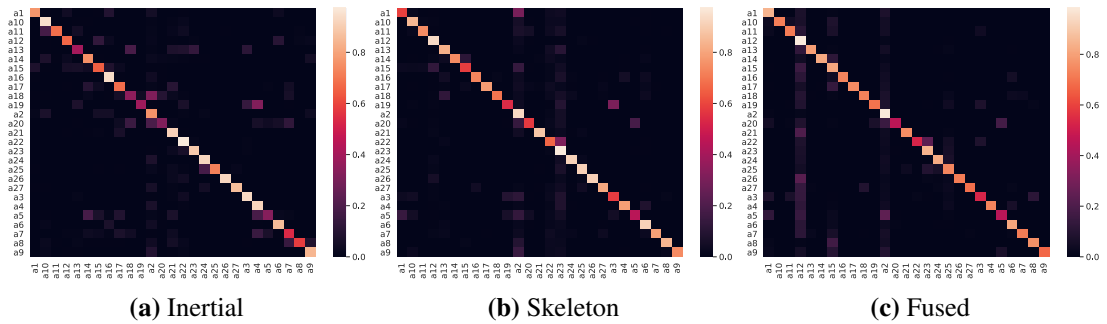


Figure 3.9: Confusion matrices for the UTD-MHAD dataset.

Table 3.4: Results on UTD-MHAD. Units are in %.

Approach	Accuracy
Zhao et al. [Zha+19b]	92.8
Wang et al. [Wan+18a]	85.8
Chen et al. (Kinect DMMs) [CJK15]	66.1
Chen et al. (Inertial) [CJK15]	67.2
Chen et al. (Fused) [CJK15]	79.1
Liu and Yuan [LY18]	94.5
<hr/>	
<i>Ours (Sparse, Skeleton)</i>	91.1
<i>Ours (Sparse, Skeleton, AIS)</i>	93.3
<i>Ours (Sparse, Inertial)</i>	72.9
<i>Ours (Sparse, Inertial, AIS)</i>	81.6
<i>Ours (Sparse, Fused)</i>	76.1
<i>Ours (Sparse, Fused, AIS)</i>	86.5
<hr/>	
<i>Ours (Dense, Skeleton)*</i>	91.6
<i>Ours (Dense, Skeleton, AIS)*</i>	91.8
<i>Ours (Dense, Inertial)*</i>	84.9
<i>Ours (Dense, Inertial, AIS)*</i>	80.5
<i>Ours (Dense, Fused)*</i>	91.5
<i>Ours (Dense, Fused, AIS)*</i>	91.9

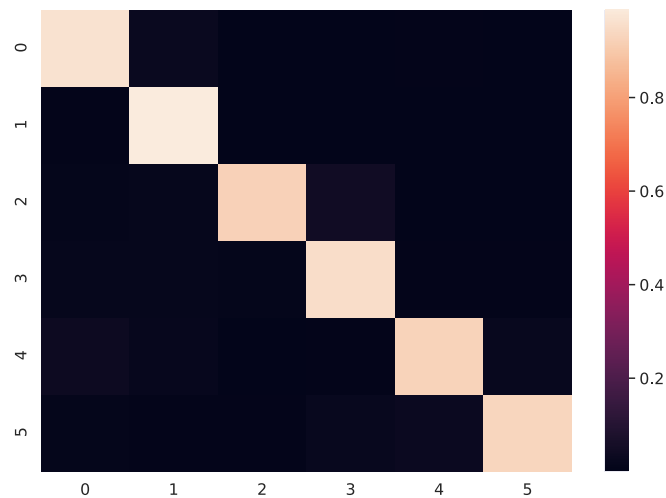


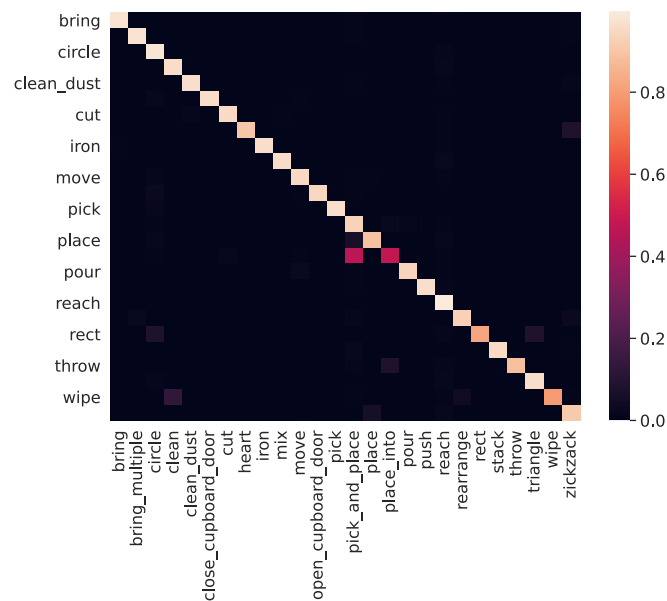
Figure 3.10: Confusion matrix for the ARIL dataset.

using the dense representation without augmentation. Our approach is outperformed by the approach from Liu and Yuan [LY18] which follows a posemap evolution method. This approach operates on additionally calculated pose features and omits the original skeleton estimates. For the inertial data, we improve the baseline by +8.8% for IMU data and +10.3% for the fusion with the proposed augmentation methods. Fusion in our experiments did not have an overall positive effect. The inertial measurements seem to negatively bias the predicted action. Additional sensor confidence encoding could guide future research. The dense representation improves over the sparse representation for the fusion and inertial experiments. Additional augmentation did only slightly improve most of the results, and for the experiments using only the inertial measurements the results were impacted negatively. In Fig. 3.9 we give the confusion matrices for the individual experiments on inertial data, skeleton data and the fusion of both on the dense representation. Interesting to note that for the fused experiments in Fig. (c) the confusion for classes *a4* and *a19* on the skeleton data (see Fig. 3.9b) and *a3* and *a19* on the inertial data (see Fig. 3.9a) are completely solved. These actions consist of fast, repeating, right hand movements (*a3*: right hand wave, *a4*: two hand front clap, *a19*: right hand knock on door).

ARIL The experiments we conducted on the ARIL dataset are compared to a 1D-ResNet CNN [Wan+19a] architecture proposed by the datasets authors. Results are presented in Table 3.5. Our approach, using the sparse representation, performs better by +3.1% and the additional proposed augmentation methods improved the baseline by +6.8%. The dense representation improves over the sparse representation by an additional improvement of 3%. Wi-Fi CSI fingerprints have the benefit of being separated

Table 3.5: Results on ARIL dataset. Units are in %.

Approach	Accuracy
Wang et al. [Wan+19a]	88.1
<i>Ours (Sparse)</i>	91.2
<i>Ours (Sparse, AIS)</i>	94.9
<i>Ours (Dense)*</i>	97.9
<i>Ours (Dense, AIS)*</i>	96.7

**Figure 3.11:** Confusion matrix for the Simitate dataset.

by their 52 bands already. Signal reduction is therefore not necessary. The additional proposed augmentation methods increase the accuracy by another 3.7% for the sparse representation. In contrast, no further enhancements were achieved by the additional augmentation on the dense representation. The highest accuracy in our experiments was achieved with the unaugmented dense representation, which improved 1D-ResNet baseline by +9.8% and over the sparse representation +6.7%. The confusion matrix shown in Fig. 3.10 supports that the approach performs well on all classes included in the dataset. This dataset is used to show our action recognition capabilities for Wi-Fi CSI fingerprints.

Table 3.6: Results on Simitate. Units are in %.

Approach	Accuracy	mpcA
Ours (Sparse, Raw)	95.72	89.15
Ours (Sparse, AIS)	96.11	90.83
Ours (Dense, Raw)*	95.96	84.57
Ours (Dense, AIS)*	93.58	85.08

Simitate On the Simitate dataset, a high accuracy is achieved on an 80/20 train/test split. Results are given in Table 3.6. We give the overall accuracy and the mean per class accuracy, as the classes are imbalanced. Augmentation of this dataset yields only a minimal improvement. This dataset is especially interesting for adding context. In addition to the hand poses, the object poses can be added by our proposed signal fusion approach. As of now, there are no comparable results published. But the results suggest applicability for symbolic imitation approaches in the future. The mean per class accuracy, denoted as mpcA is generally lower, and demonstrates that classes with a lower number of samples have a slightly lower performance. In our experiments, the sparse representation yields in general a higher mpcA, suggesting that this representation performs better for underrepresented classes. In this experiment, the augmentation has a positive effect on the sparse representation, while results are negatively influenced for the dense representation. The number of motion capturing markers tracked are lower than the skeleton joints and depending on the number of interacting objects are similar to the number of inertial axes. The overall best result is achieved for an unaugmented sparse representation. However, this representation is not performing significantly better than the dense counterpart.

ETRI-Activity-3D The ETRI-Activity-3D and the following datasets are included to show generalization capabilities over various applications. For the following datasets, we focus on experiments with the dense representation. Results for our experiments with the ETRI-Activity-3D dataset are given in Table 3.7 and the confusion matrix for the dense representation in Fig. 3.12. Our approach yields en-par results with the four stream [Jan+20] approach proposed by the dataset authors, with additional rotation augmentation our approach reaches *state-of-the-art* performance on the ETRI-Activity-3D dataset. Only skeleton sequences in a single stream have been used in our experiments, suggesting superior performance of the simple EfficientNet architecture over more complex multi-stream architectures that simultaneously train additional spatial, short- and long-term temporal feature extractors. The most confused classes are similar classes 28 (washing a towel by hands) and 11 (washing hands) 22 (washing dishes). Note, our

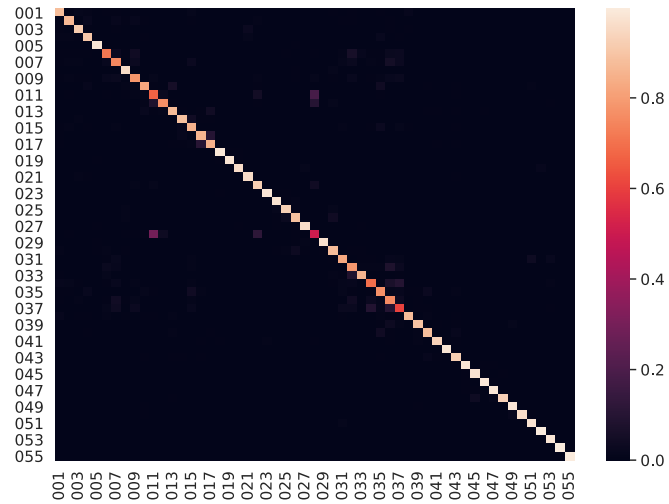


Figure 3.12: Confusion matrix for the ETRI-Activity-3D dataset.

Approach	Accuracy [%]
Beyond Joint [WW18]	79.1
SK-CNN [Cao+19]	83.6
ST-GCN [YXL18]	86.8
Motif ST-GCN [Wen+19]	89.9
Ensem-NN [Xu+18]	83.0
MANs [Liu+21a]	82.4
HCN [Li+18]	88.0
FSA-CNN [Jan+20]	90.6
<i>Ours (dense)</i>	90.7
<i>Ours (dense, AIS)</i>	91.1

Table 3.7: Action recognition results on the ETRI-Activity-3D dataset.

approach encodes only skeleton sequences. To distinguish between those classes additional visual clues, to encode the presence and type of interacting object, could be fused into the representation for further improvement.

Toyota Smarthome A confusion matrix of a complete model trained with the dense representation after 150 epochs is given in Fig. 3.13. Reasonable errors appear with the similar subclasses of e.g., *cooking*, *drinking*, and *eating* as the representation encodes

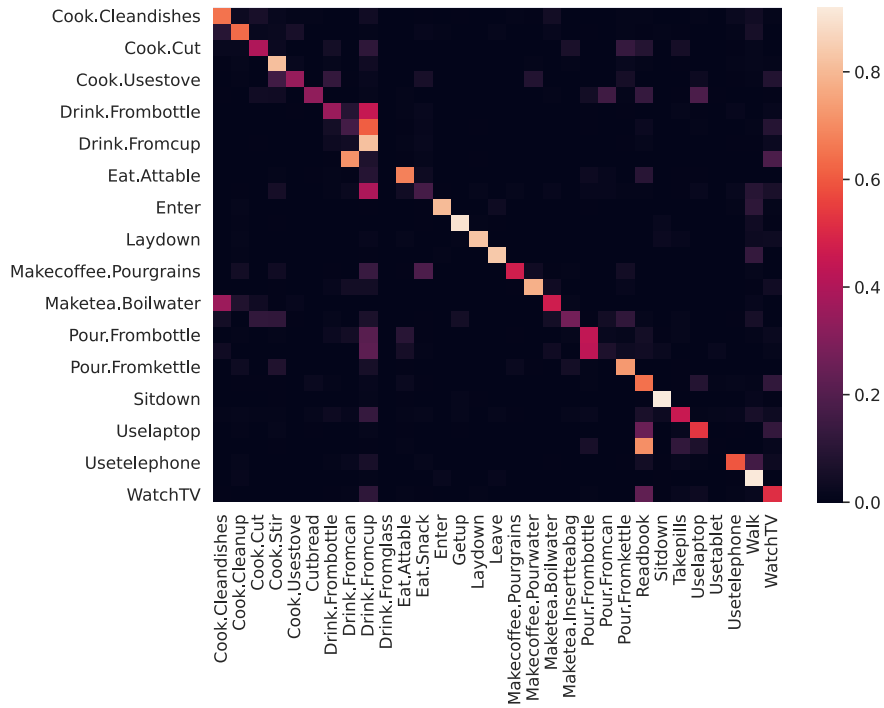


Figure 3.13: Toyota Smarthome, Dense Representation, 150 epochs.

Approach	CS mpcA [%]
Dense Trajectories [SYS14]	41.9
LSTM [MT16]	42.5
<i>Ours (Dense)</i>	53.2
I3D [CZ17]	53.4
I3D [CZ17] + NL [Wan+18c]	53.6
Separable-STA [Das+19]	54.2
AssembleNet++[Ryo+20]	63.6

Table 3.8: Toyota Smarthome action recognition results. mpcAs are given.

only joint movement and lacks information about the interacting objects.

Results are given in Table 3.8. As the dataset is unscripted, the number of class samples per class has a high variance, therefore the mean per-class accuracy is proposed as a metric for comparison. Our approach, despite using only skeleton sequences as input, outperforms the LSTM-based approach by Mahasseni and Todorovic [MT16] which also utilizes only the skeleton sequences and the dense trajectory approach which relies

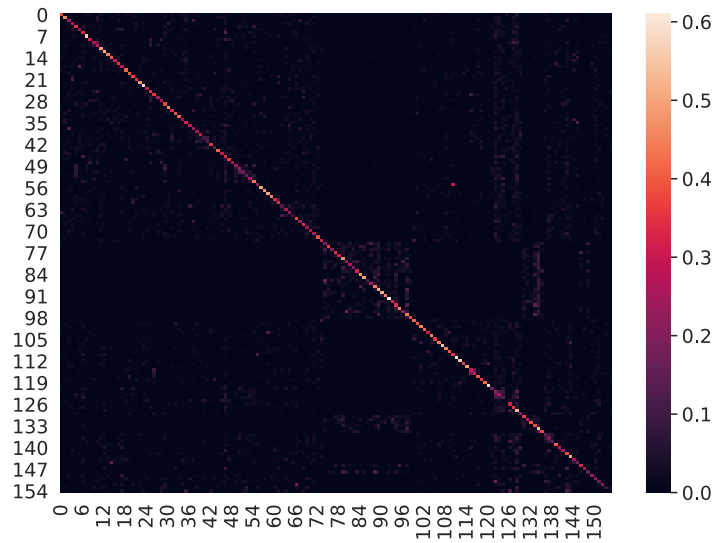


Figure 3.14: Confusion matrix for the UAV-Human dataset, using the dense representation, after 150 epochs, with additional augmentation.

on densely matched features [MT16]. Our approach yields comparable performance to the inflated 3D-CNN approach [CZ17], which operates on the video sequences. The Separable-STA [Das+19] approach improves the I3D approach by additionally guiding the training process with skeleton sequences. The AssembleNet++ [Ryo+20] approach operates on RGB sequences and aims at learning interactions between the objects and the other raw inputs with a supporting peer-attention module. Also, for these experiments, the encoding of additional visual clues for the interacting objects would be required for further enhancements.

UAV-Human Results using the dense representation after a training of 150 epochs are given in Table 3.9 and the corresponding confusion matrix for the augmented results in Fig. 3.14. Note, the legend scales only to approximately 0.6, highlighting that none of the classes achieve a particular high per class accuracy. In the experiments, we focus on the CSv1 split, which has a higher variance in the execution of the actions for testing an training. Multiple similar classes show increased confusion. Also interesting to highlight is the near-center block in the confusion matrix, where the action classes (74 - 98) for multi-person actions are contained. Our approach is separating these classes quite well from other classes. We experiment with single-person representations and two-person representations for actions where multiple people interact. Our approach outperforms the ST-GCN [YXL18] approach but performs worse than recent improve-

Approach	Accuracy (CSv1) [%]
DGNN [Shi+19a]	29.90
ST-GCN [YXL18]	30.25
<i>Ours (Dense)</i>	31.73
<i>Ours (Dense, AIS)</i>	33.45
<i>Ours (Dense, 2 persons)</i>	32.90
<i>Ours (Dense, 2 persons, AIS)</i>	34.40
2S-AGCN [Shi+19b]	34.84
Hard-Net [Li+20b]	36.97
Shift-GCN [Che+20]	37.98

Table 3.9: Action recognition results for the UAV-Human dataset.

Table 3.10: Action recognition results on the Kinetics-400 dataset. Top 1 and Top 5 accuracies are given in %.

Approach	Top 1	Top 5
Feature Enc. [Fer+15]	14.9	25.8
Deep LSTM [Sha+16]	16.4	35.3
Temporal Conv. [KR17]	20.3	40.0
<i>Ours (Dense)</i>	22.7	41.7
<i>Ours (Dense, AIS)</i>	24.1	43.1
ST-GCN [YXL18]	30.7	52.8
2s-AGCN [Shi+19b]	36.1	58.7

ments to the ST-GCN approach [Shi+19b; Li+20b; Che+20].

Kinetics 400 Results for the Kinetics 400 dataset, using the pre-generated pose feature from Yan et al. [YXL18], are shown in Table 3.10. This dataset is especially challenging as it has 400 different action classes and consists of highly varying videos sourced from YouTube. The results reported on this dataset are reported for the dense representation for 150 epochs. Augmentation improved the results by +1.4%. Our approach performs significantly better than the LSTM-based approach [Sha+16], and the temporal-convolution based approach [KR17]. However, GCN-based approaches perform significantly better than our CNN based approach. Our approach is outperformed

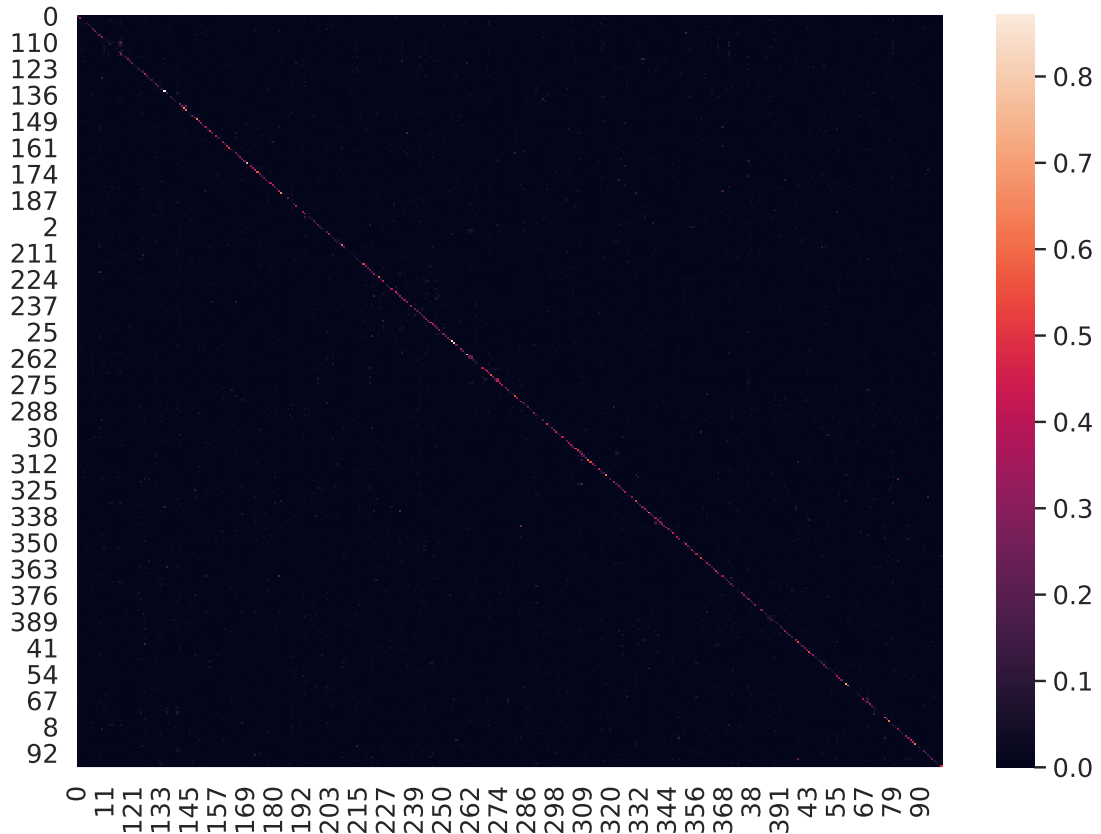


Figure 3.15: Kinetics-400 confusion matrix.

by the ST-GCN [YXL18] and the 2s-AGCN [Shi+19b] approaches, suggesting that the GCN-based model scales better with larger amounts of classes. Samples from the Kinetics 400 dataset have large variances in their sequence length, this is currently a constraint of our approach as we scale the input sequences to a fixed length to pass it then to the CNN. The confusion matrix is shown in Fig. 3.15. Visual cues could also improve the action recognition accuracy. All the reported results, from Table 3.10, have the same data-basis and therefore none of the listed approaches has visual information encoded.

Sensor Modality Generalization Table 3.11 gives results for different modalities and different datasets relating to other ex excerpt of related methods. Our approach achieves good accuracies across the different datasets and different splits. It does not necessarily compete with recent GCN-based approaches for skeleton-based action recognition, but competes very well with CNN-based action recognition methods, while still generalizing well to various other sensor modalities. Our approach is the only one adapting to

Table 3.11: Action recognition results on four different datasets. Accuracy in [%] is given.

Approach	Type	NTU 60		NTU 120		UTD-MHAD				Simitate	ARIL	#
		CS	CV	CS	CV	RGB	Skl	IMU	Fused	MoCap	Wi-Fi	
<i>Ours (Sparse)</i>	CNN	-	-	70.8	71.6	-	93.3	81.6	86.5	96.1	94.9	4
<i>Ours (Dense)</i>	CNN	83.3	81.7	80.0	82.0	-	93.9	80.6	96.0	96.0	97.9	4
Imran et al. [IR20]	CNN+RNN	-	-	-	-	83.5	93.5	86.5	97.9	-	-	3
Ehatisham et al. [Eha+19]	HOG	-	-	-	-	85.2	-	91.6	98.3	-	-	2
Liu et al. [Liu+16a]	LSTM	69.2	77.7	55.7	57.9	-	-	-	-	-	-	1
Liu et al. [LLC17]	CNN	80.0	87.2	60.3	63.2	-	-	-	-	-	-	1
Liu et al. [Liu+20b]	GCN	91.5	96.2	86.9	88.4	-	-	-	-	-	-	1
Wang et al. [Wan+19a]	CNN	-	-	-	-	-	-	-	-	-	89.6	1

generalize across 4 different sensor modalities without major adoptions. For the UTD-MHAD we got the highest accuracy on skeleton sequences, and improve by a high margin over the fused accuracy with the dense representation. Individual architectures per modality potentially lead to higher recognition accuracies [IR20; Eha+19]. However, we claim that our approach simplifies the action recognition training and inference by a common architecture for all modalities and relieve the need for individual streams per modality. For motion capturing experiments, we compared sparse and dense representation, that perform comparably well. Similar to the Wi-Fi experiments, we perform better as the baseline 1D-ResNet CNN approach by Wang et al. [Wan+19a] and perform comparably well to the augmented results of sparse representation. For the fusion experiments, we decided to use an early fusion method to avoid multiple network-streams to be trained individually. Fusion is done by concatenating the signal matrices after sub-sampling the higher frequent modality. The fusion with the dense representation performs much better than the fusion with the sparse representation. For the Simitate dataset, we could add object context by fusing the interacting objects to the hand pose measurements. A late fusion method might improve the fusion, however will add complexity to the overall model by introducing individual network streams. Our approach mostly benefits by its simplicity and wide variety of supported modalities over the current available action recognition approaches. Our approach can not compete directly with the most recent approaches for skeleton-based action recognition like [Shi+19b], but generalizes across various modalities. Further, our approach still achieves a quite high accuracy for both the cross-view and cross-setup accuracy, even outperforming the earlier graph convolutional neural networks [Pap+20; YXL18] on some experiments.

Discussion Most approaches focus on getting high accuracy on a single modality, whereas our approach on a signal level serves as an interesting framework for multi-modal action recognition. In total, we have shown good results across 4 modalities (Skeleton, IMU, Motion Capturing System, Wi-Fi). To the authors' knowledge, no experiment with a similar extent is known. A benefit is the common representation that allows immediate prototyping. Run times are constant, even when additional context or sensors are added due to the representation level fusion. The EfficientNet-B2 architecture serves as a good basis for action recognition on our representation. Additional augmentation has improved the accuracy across the conducted experiments.

3.5 Conclusion

This chapter proposes to transform individual signals of different sensor modalities and represent them as an image, either with a sparse or dense representation. The resulting images are then classified using an EfficientNet-B2 architecture. The signal level formulation has shown to be sufficient for generalization across various sensor modalities

for action recognition. This contrasts with many previously proposed approaches that often focus on action recognition on a single modality. Our approach was evaluated on action recognition datasets based on skeleton estimates, inertial measurements, motion capturing data and Wi-Fi CSI fingerprints on a wide set of public available datasets. To the best of our knowledge, there is no other approach contesting such a broad generalization across various sensor data modalities. For skeleton data, we represent each joint and their respective axis as individual signals. For Wi-Fi, we used each of the 52 CSI fingerprint channels as signals. For inertial measurement units, we used each axis of the acceleration and angular velocity. For our motion capturing experiments, we used each axis of the marker attached to the hand and the interacting objects. Additional context like subjects and object estimates or even the fusion of different modalities can be flexibly added by a matrix concatenation. As our approach is limited to sparse signals, we propose filtering methods on a signal level to reduce signals that do not contribute much to the action. By this, additional information can be added without overloading the image representation. We evaluated our approach on four different datasets: the NTU 120 dataset for skeleton data, the UTD-MHAD dataset for skeleton and inertial data, the ARIL dataset for Wi-Fi data and the Simitate dataset for motion capturing data. To show better generalization across various applications, we also experimented with the Toyota Smarthome, the ETRI-Activity-3D, the Kinetics 400 and the UAV-Human dataset.

We found that our approach achieves action recognition en-par with current state-of-the-art approaches on datasets like the ETRI-Activity-3D. With the introduction of many more classes like up to 400 on the Kinetics-400 dataset, our approach is outperformed by recent GCN-based approaches for skeleton-based action recognition. However, our approach targets better generalization across various sensor modalities in the first place. The experimental results show that our approach is achieving good results across the different sensor modalities.

Chapter 4

Multi-Modal Action Recognition using Graph Convolutional Networks

In contrast to the previous chapter (Chapter 3), which focused on action recognition on various sensor data modalities using CNNs, this chapter presents *Fusion-GCN*, an approach for multimodal action recognition using a GCN.

GCNs have recently shown *state-of-the-art* performance for skeleton-based action recognition, but are currently widely neglected as the foundation for the fusion of various sensor modalities. Therefore, we propose to incorporate additional modalities like IMU or RGB features into a skeleton-graph, either on a channel- or spatial dimension. On a channel dimension, modalities are fused by introducing additional node attributes. On a spatial dimension, additional nodes are incorporated to the skeleton-graph. We evaluated our approach on two publicly available datasets: the UTD-MHAD dataset and the MMAct dataset. Most notably, *Fusion-GCN* improves the current baseline on the MMAct dataset significantly with the fusion of skeleton-estimates and accelerometer measurements from a smart-watch. We argue that *Fusion-GCN* can influence future fusion approaches on the basis of graph convolutional neural networks.

This chapter is based on our work presented in [DMP21]. This publication was written in cooperation with Michael Duhme and represents an improved version of his master thesis supervised by me. Michael Duhme contributed significantly to the technical realization, the experiments and the writing. I contributed significantly to the research idea and the literature review.

4.1 Introduction

Automatic Human Action Recognition (HAR) is a research area that is utilized in various fields of application where human monitoring is infeasible due to the amount of data and scenarios where quick reaction times are vital, such as surveillance and real-

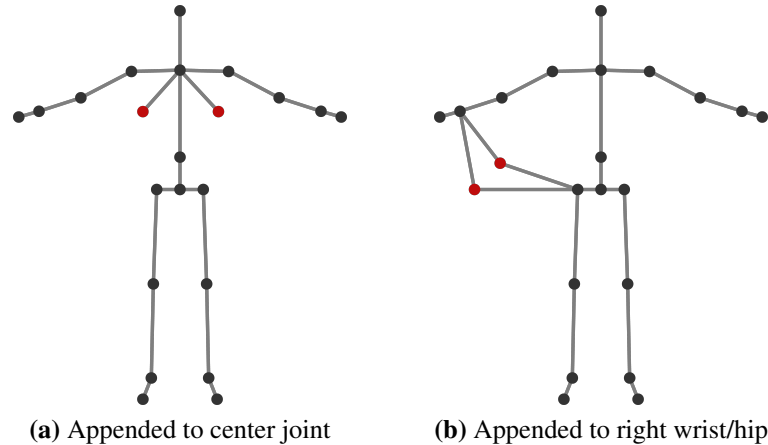


Figure 4.1: Showing the skeleton as included in UTD-MHAD. IMU nodes are either appended to the central node (neck) or to both the right wrist and right hip. Two additional representations arise when all newly added nodes are themselves connected by edges.

time monitoring of suspicious and abnormal behavior in public areas [Niu+04; Hu+07; NYK09; TJA18] or intelligent hospitals and healthcare sectors [Gao+18; Duo+05] with scenarios such as fall detection [Nou+07; ST17], detection of potentially life-threatening situations [Duo+05], and monitoring of medication intake [Huy+09]. Additional areas of applications include video retrieval [RY16], robotics [Ryo+15], smart home automation [Kot+19], and autonomous vehicles [ZBX18]. In recent years, approaches based on neural networks, especially GCNs like ST-GCN [YXL18] or 2s-AGCN [Shi+19b], have achieved state-of-the-art results in classifying human actions from skeleton sequences.

GCNs can be seen as an extension to CNNs that work on graph-structured data [KW17]. Its network layers operate by including a binary or weighted adjacency matrix, that describes the connections between each of the individual graph nodes. As of now, due to their graph-structured representation in the form of joints (graph nodes) and bones (graph edges), research for HAR using GCNs is mainly limited to skeleton-based recognition. However, the fusion of additional modalities into GCNs models are currently neglected. For that reason, taking skeleton-based action recognition as the foundation, our objective is to research possibilities of incorporating other vision-based modalities and modalities from worn sensors into existing GCN models for skeleton-based action recognition through data fusion and augmentation of skeleton sequences. Fig. 4.1 provides an example of two suggestions on how inertial measurements can be incorporated into a skeleton graph. To the best of our knowledge, Fusion-GCN is the first approach proposing to flexibly incorporate additional sensor modalities into the skeleton graph for HAR. We evaluated our approach on two multimodal datasets, UTD-MHAD [CJK15] and MMAAct [Kon+19].

The contributions of this chapter can be summarized as:

- We propose the fusion of multiple modalities by incorporating sensor measurements or extracted features into a graph representation. The proposed approach significantly lifts the state-of-the-art on the large-scale MMAAct dataset.
- We propose modality fusion for GCNs on two dimensionality-levels: (a) the fusion at a channel dimension to incorporate additional modalities directly into the already existing skeleton nodes, (b) the fusion at a spatial dimension to incorporate additional modalities as new nodes spatially connected to existing graph nodes.
- We demonstrate the applicability of the flexible fusion for various modalities like skeleton, inertial, RGB data in an early fusion approach.

4.2 Related Work

In this section, we present related work from the skeleton-based action recognition domain that is based on GCN and further present recent work on multi-modal action recognition.

Skeleton-based Action Recognition Approaches based on GCNs have recently shown great applicability on non-Euclidean data [Pen+20] like naturally graph-structure represented skeletons and have recently defined the state-of-the-art. Skeletons, as provided by large-scale datasets [Sha+16], commonly are extracted from depth cameras [Sho+11]. RGB images can be transformed into human pose feature that yield a similar skeleton-graph in 2D [Cao+21; KBA19; Lug+19] and in 3D [Lug+19; Meh+20; Iqb+18]. All of those approaches output skeleton-graphs that are suitable as input for our fusion approach as a base structure for the incorporation of additional modalities. The Spatial-Temporal Graph Convolutional Network (ST-GCN) [YXL18] is one of the first proposed models for skeleton-based HAR that utilizes GCNs based on the propagation rule introduced by Kipf and Welling [KW17]. The Adaptive Graph Convolutional Network (AGCN) [Shi+19b] builds on these fundamental ideas with the proposal of learning the graph topology in an end-to-end-manner. Peng et al. [Pen+20] propose a Neural Architecture Search (NAS) approach for finding neural architectures to overcome the limitations of GCN caused by fixed graph structures. Cai et al. [Cai+21] proposes to add flow patches to handle subtle movements into a GCN. Approaches based on GCN [Che+20; Pap+20; Son+20a; Li+19a] have been constantly improving the state-of-the-art on skeleton-based action recognition recently.

Multi-Modal Action Recognition Chéron et al. [CLS15] design CNN input features based on the positions of individual skeleton joints. Here, human poses are applied to RGB images and optical flow images. The pixel coordinates that represent skeleton joints are then grouped hierarchically starting from smaller body parts, such as arms, and upper body to full body. For each group, an RGB image and optical flow patch is cropped and passed to a 2D-CNN. The resulting feature vectors are then processed and concatenated to form a single vector, which is used to predict the corresponding action label. Similarly, Cao et al. [Cao+16] propose to fuse pose-guided features from RGB-Videos. Cao et al. [Cao+18] further, refine this method by using different aggregation techniques and an attention model. Islam and Iqbal [II20] propose to fuse data of RGB, skeleton and inertial sensor modalities by using a separate encoder for each modality to create a similar shaped vector representation. The different streams are fused using either summation or vector concatenation. With Multi-GAT [II21] an additional message-passing graphical attention mechanism was introduced. Li et al. [Li+20a] propose another architecture that entails skeleton-guided RGB features. For this, they employ ST-GCN to extract a skeleton feature vector and R(2+1)D [Tra+18] to encode the RGB video. Both output features are fused either by concatenation or by compact bilinear correlation.

The above-mentioned multi-modal action recognition approaches follow a late-fusion method, that fuse various models for each modality. This allows a flexible per modality model-design, but comes at the computational cost of the multiple streams that need to be trained. For early fusion approaches, multiple modalities are fused on a representation level [MTP20a], reducing the training process to a single model but potentially loosing the more descriptive features from per-modality models. Kong et al. [Kon+19] presented a multi modality distillation model. Teacher models are trained separately using a 1D-CNN. The semantic embeddings from the teaching models are weighted with an attention mechanism and are ensembled with a soft target distillation loss into the student network. Similarly, Liu et al. [Liu+21b] utilize distilled sensor information to improve the vision modality. Luo et al. [Luo+18] propose a graph distillation method to incorporate rich privileged information from a large-scale multi-modal dataset in the source domain, and improves the learning in the target domain. More fundamentally, multi-modality in neural networks is recently also tackled by the multi-modal neurons that respond to photos, conceptual drawings and images of text [Goh+21]. Joze et al. [Joz+20] propose a novel intermediate fusion scheme in addition to early and late-fusion, they share intermediate layer features between different modalities in CNN streams. Perez-Rua et al. [Per+19] presented an approach for finding neural architecture search for the fusion of multiple modalities. To the best of our knowledge, our Fusion-GCN approach is the first that proposes to incorporate additional modalities directly into the skeleton-graphs as an early fusion scheme.

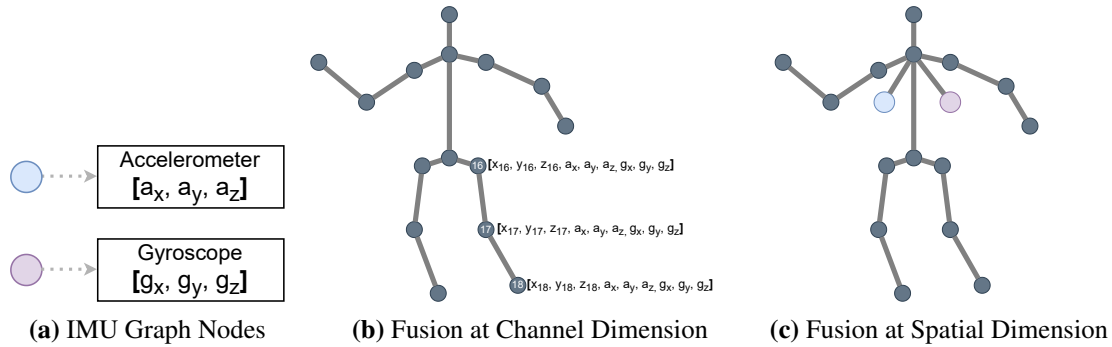


Figure 4.2: Options for fusion of skeleton graph and IMU signal values, viewed as skeleton nodes. If both skeleton joint coordinates and wearable sensor signals share a common channel dimension, the skeleton graph can be augmented by simply appending signal nodes at some predefined location.

4.3 Approach

In the context of multi-modal action recognition, early and late fusion methods have been established to either fuse on a representation or feature level. We present approaches for fusion of multiple modalities at representation level to create a single graph which is passed to a GCN.

4.3.1 Incorporating Additional Modalities Into a Graph Model

Early fusion denotes the combination of structurally equivalent streams of data before sending them to a larger (GCN) model, whereas late fusion combines resulting outputs of multiple neural network models. For early fusion, one network handles multiple data sources which are required to have near identical shape to achieve fusion. As done by Song et al. [Son+18], each modality may be processed by some form of an encoder to attain a common structure before being fused and passed on to further networks. Following a skeleton-based approach, for example, by employing a well established GCN model like ST-GCN or AGCN as the main component, RGB and inertial measurements are remodeled and factored into the skeleton structure. With Fusion-GCN we suggest the flexible integration of additional sensor modalities into a skeleton graph by either adding additional node attributes (*fusion on a channel dimension*) or introducing additional nodes (*fusion at a spatial dimension*). In detail, the exact possible fusion approach is as follows.

Let $\mathbf{X}_{\text{Skl}} \in \mathbb{R}^{(M \times C_{\text{Skl}} \times T_{\text{Skl}} \times N_{\text{Skl}})}$ be a skeleton sequence input, where M is the number of actors that are involved in an action, C_{Skl} is the initial channel dimension (2D or 3D joint coordinates) and sizes T_{Skl} and N_{Skl} are sequence length and

number of skeleton graph nodes. An input of shape $\mathbb{R}^{(M \times C \times T \times N)}$ is required when passing data to a spatial-temporal GCN model, such as ST-GCN. Furthermore, let $\mathbf{X}_{\text{RGB}} \in \mathbb{R}^{(C_{\text{RGB}} \times T_{\text{RGB}} \times H_{\text{RGB}} \times W_{\text{RGB}})}$ be the shape of an RGB video with channels C_{RGB} , frames T_{RGB} and image size $H_{\text{RGB}} \times W_{\text{RGB}}$. For sensor data, the input is defined as $\mathbf{X}_{\text{IMU}} \in \mathbb{R}^{(M \times C_{\text{IMU}} \times S_{\text{IMU}} \times T_{\text{IMU}})}$, where T_{IMU} is the sequence length, S_{IMU} is the number of sensors and C_{IMU} is the channel dimension. For example, given gyroscope and accelerometer with x-, y- and z-values each, the structure would be $S_{\text{IMU}} = 2$ and $C_{\text{IMU}} = 3$. Similar to skeleton data, M denotes the person wearing the sensor and its value is equivalent to that of skeleton, that is, $M_{\text{Skl}} = M_{\text{IMU}}$. Considering a multi-modal model using a skeleton-based GCN approach, early fusion can now be seen as a task of restructuring non-skeleton modalities to be similar to skeleton sequences by finding a mapping $\mathbb{R}^{(C_{\text{RGB}} \times T_{\text{RGB}} \times H_{\text{RGB}} \times W_{\text{RGB}})} \rightarrow \mathbb{R}^{(M \times C \times T \times N)}$ or $\mathbb{R}^{(M \times C_{\text{IMU}} \times S_{\text{IMU}} \times T_{\text{IMU}})} \rightarrow \mathbb{R}^{(M \times C \times T \times N)}$ with some C , T and N . This problem can be reduced: If the sequence length of some modalities is different, $T_{\text{Skl}} \neq T_{\text{RGB}} \neq T_{\text{IMU}}$, a common T can be achieved by resampling T_{RGB} and T_{IMU} to be of the same length as the target modality T_{Skl} . Early fusion is then characterized by two variants of feature concatenation to fuse data:

1. Given \mathbf{X}_{Skl} and an embedding $\mathbf{X}_E \in \mathbb{R}^{(M \times C_E \times T \times N_E)}$ with sizes C_E and N_E where $N = N_{\text{Skl}} = N_E$, fusion at the channel dimension means creating a fused feature $\mathbf{X} \in \mathbb{R}^{(M \times C_{\text{Skl}} \times C_E \times T \times N)}$. An example is shown in Figure 4.2b.
2. Given an embedding where $C = C_{\text{Skl}} = C_E$ instead, a second possibility is fusion at the spatial dimension, that is, creating a feature $\mathbf{X} \in \mathbb{R}^{(M \times C \times T \times N_{\text{Skl}} + N_E)}$. Effectively, this amounts to producing $M \cdot T \cdot N_E$ additional graph nodes and distributing them to the existing skeleton graph at each time step by resizing its adjacency matrix and including new connections. In other words, the already existing skeleton graph is extended by multiple new nodes with an identical number of channels. An example is shown in Figure 4.2c.

The following sections introduce multiple approaches for techniques about the early fusion of RGB video and IMU sensor modalities together with skeleton sequences by outlining the neural network architecture.

4.3.2 Fusion of Skeleton Sequences and RGB Video

This section explores possibilities for fusion of skeleton sequences and 2D data modalities. Descriptions and the following experiments are limited to RGB video, but all introduced approaches are in the same way applicable to depth sequences. As previously established, early fusion of RGB video and skeleton sequences in preparation for a skeleton-based GCN model is a problem of finding a mapping $\mathbb{R}^{(C_{\text{RGB}} \times H_{\text{RGB}} \times W_{\text{RGB}})} \rightarrow \mathbb{R}^{(M \times C \times N)}$. An initial approach uses a CNN to compute vector representations of

$N \cdot M \cdot T$ skeleton-guided RGB patches that are cropped around projected skeleton joint positions. Inspired by the work of Wang and Gupta [WG18] and Norcliffe-Brown et al. [NVP18], a similar approach involves using an encoder network to extract relevant features from each image of the RGB video. This way, instead of analyzing $N \cdot M \cdot T$ cropped images, the T images of each video are utilized in their entirety. A CNN is used to extract features for every frame and fuse the resulting features with the corresponding skeleton graph, before the fused data is forwarded to a GCN. By running this procedure as part of the training process and performing fusion with skeleton sequences, the intention is to let the encoder network extract those RGB features that are relevant to the skeleton modality. For example, an action involving an object cannot be fully represented by merely the skeleton modality because an object is never part of the extracted skeleton. Objects are only visible in RGB video.

4.3.3 Fusion of Skeleton Sequences and IMU Signals

Fusion of skeleton and data from wearable sensors, such as IMUs, is applicable in the same way as described in the fusion scheme from the previous section. In preparation to fuse both modalities, they again need to be adjusted to have an equal sequence length first. Then, assuming both the skeleton joint coordinates and the signal values have a common channel dimension $C = 3$ and because $M_{\text{Skl}} = M_{\text{IMU}}$, since all people wear a sensor, the only differing sizes between skeleton modality and IMU modality are N , the number of skeleton graph nodes, and S , the number of sensor signals. Leaving aside its structure, the skeleton graph is a collection of N nodes. A similar understanding can be applied to the S different sensors. They can be understood as a collection of S graph nodes (see Figure 4.2a). The fusion of sensor signals with the skeleton graph is therefore trivial because the shape is almost identical. According to channel dimension fusion as described in the previous section, the channels of all S signals can be broadcasted to the x-, y- and z-values of all N skeleton nodes to create the GCN input feature $\mathbf{X} \in \mathbb{R}^{(M \times (1+S) \cdot C \times T \times N)}$, as presented in Figure 4.2b. The alternative is to append all S signal nodes onto the skeleton graph at some predefined location to create the GCN input feature $\mathbf{X} \in \mathbb{R}^{(M \times C \times T \times (N+S))}$, as illustrated in Figure 4.2c. Similar to the RGB fusion approaches, channel dimension fusion does not necessarily require both modalities to have the same dimension C if vector concatenation is used. In contrast, the additional nodes are required to have the same dimension as all existing nodes if spatial dimension fusion is intended.

4.3.4 Combining Multiple Fusion Approaches

All the introduced fusion approaches can be combined into a single model, as illustrated by Figure 4.3. First, the RGB modality needs to be processed using one of the

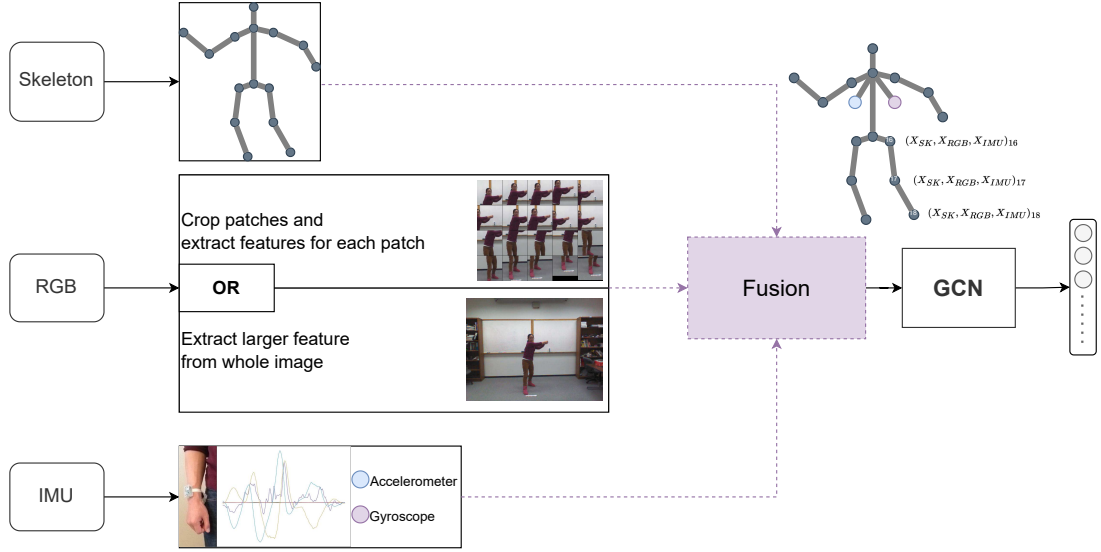


Figure 4.3: All described approaches can be flexible fused together for early fusion and passed to a GCN. Fusion can be realized independent of a channel or spatial fusion dimension. Here we give an example of a mixed (channel and spatial) fusion.

variants discussed in Section 4.3.2. Ideally, this component runs as part of the supervised training process to allow the network to adjust the RGB feature extraction process based on the interrelation of its output with the skeleton graph. Similarly, sensor signals need to be processed using one of the variants discussed previously for that modality. Assuming all sequences are identical in length, to combine the different representations, let $\mathbf{X}_{\text{Skl}} \in \mathbb{R}^{(M \times C_{\text{Skl}} \times T \times N_{\text{Skl}})}$ be the sequence of skeleton graphs. For RGB, let $\mathbf{X}_{\text{RGB1}} \in \mathbb{R}^{(M \times C_E \times T \times N)}$ be the C_E -sized channel features obtained from computing individual patch features or feature extraction for the whole image or $\mathbf{X}_{\text{RGB2}} \in \mathbb{R}^{(M \times C \times T \times N_E)}$ be the RGB feature representing additional graph nodes. Respectively, the two variants of generated IMU features are $\mathbf{X}_{\text{IMU1}} \in \mathbb{R}^{(M \times S \cdot C_{\text{IMU}} \times T \times N)}$ or $\mathbf{X}_{\text{IMU2}} \in \mathbb{R}^{(M \times C_{\text{IMU}} \times T \times S)}$. The following possibilities to fuse different combinations of these representations arise.

- $(\mathbf{X}_{\text{Skl}}, \mathbf{X}_{\text{RGB1}}, \mathbf{X}_{\text{IMU1}}) \rightarrow \mathbf{X}_{\text{Fused}} \in \mathbb{R}^{(M \times C_{\text{Skl}} + C_E + S \cdot C_{\text{IMU}} \times T \times N_{\text{Skl}})}$ is the feature when combining modalities at channel dimension by vector concatenation.
- $(\mathbf{X}_{\text{Skl}}, \mathbf{X}_{\text{RGB1}}, \mathbf{X}_{\text{IMU2}}) \rightarrow \mathbf{X}_{\text{Fused}} \in \mathbb{R}^{(M \times C_{\text{Skl}} + C_E \times T \times N_{\text{Skl}} + S)}$ combines skeleton with computed RGB features at channel dimension and expands the skeleton graph by including additional signal nodes. Since $C_{\text{IMU}} = C_{\text{Skl}}$, the newly added nodes also need to be extended to have $C_{\text{Skl}} + C_E$ channels. In contrast to skeleton nodes, there exists no associated cropped patch or RGB value. Therefore, the remaining C_E values can be filled with zeros. Conversely, the same applies when

replacing \mathbf{X}_{RGB1} with \mathbf{X}_{RGB2} and \mathbf{X}_{IMU2} with \mathbf{X}_{IMU1} .

- $(\mathbf{X}_{\text{Skl}}, \mathbf{X}_{\text{RGB2}}, \mathbf{X}_{\text{IMU2}}) \rightarrow \mathbf{X}_{\text{Fused}} \in \mathbb{R}^{(M \times C \times T \times N_{\text{Skl}} + N_E + S)}$ introduces new nodes for both RGB and signal modalities. This is accomplished by appending them to a specific location in the graph.

4.4 Experiments

We conducted experiments on two public available datasets and various modality fusion experiments. If not stated otherwise we use the top-1 accuracy as reporting metric for the final epoch of the trained model.

4.4.1 Datasets

We now present the datasets used for the evaluation of the Fusion-GCN approach. The UTD-MHAD dataset is a great candidate for showing concepts and the MMAct dataset for generalization on a larger set of samples and more complex actions.

UTD-MHAD The UTD-MHAD dataset has been used in Chapter 3 as well. For better readability we introduce briefly re-introduce the dataset here again. UTD-MHAD [CJK15] is a relatively small dataset containing 861 samples and 27 action classes, which thereby results in shorter training durations for neural networks. Eight individuals (four females and four males) perform each action a total of four times, captured from a front-view perspective by a single Kinect camera. UTD-MHAD also includes gyroscope and accelerometer modalities by letting each subject wear the inertial sensor on either the right wrist or on the right hip, depending on whether an action is primarily performed using the hands or the legs. For the following experiments using this dataset, the protocol from the original paper [CJK15] is used.

MMAct The MMAct dataset [Kon+19] contains more than 35k data samples and 35 available action classes. With 20 subjects and four scenes with four currently available different camera perspectives each, the dataset offers a larger variation of scenarios. RGB videos are captured with a resolution of 1920×1080 pixels at a frame rate of 30 frames per second. For inertial sensors, acceleration, gyroscope and orientation data is obtained from a smartphone carried inside the pocket of a subject’s pants. Another source for acceleration data is a smartwatch, resulting in data from four sensors in total. For the following experiments using this dataset, the protocol from the original paper [Kon+19] is used which proposes a cross-subject and a cross-view split. Since skeleton sequences are missing in the dataset, we create them from RGB data using OpenPose [Cao+21].

4.4.2 Implementation

Models are implemented using PyTorch 1.6 and trained on a Nvidia RTX 2080 GPU with 8GB of video memory. To guarantee a deterministic and reproducible behavior, all training procedures are initialized with a fixed random seed. Unless stated otherwise, experiments regarding UTD-MHAD use a cosine annealing learning rate scheduler [LH17] with a total of 60 epochs, warm restarts after 20 and 40 epochs, an initial learning rate of $1e-3$ and ADAM [KB15] optimization. Experiments using RGB data instead run for 50 epochs without warm restarts. Training for MMAcT adopts the hyperparameters used by Shi et al. [Shi+19b]. For the MMAcT, skeleton and RGB features were extracted for every third frame for more efficient pre-processing and training. The base GCN model is a single-stream AGCN for all experiments.

4.4.3 Comparison to the State-of-the-Art

We now compare our Fusion-GCN approach with results reported in recent literature for each of the two datasets. For both datasets we report the accuracy. For the MMAcT dataset, we additionally report the F1-Measure, as described in Section 2.3, to be aligned with the original dataset metric.

UTD-MHAD Table 4.1 shows a ranking of all conducted experiments in comparison with other recent state-of-the-art techniques that implement multimodal HAR on UTD-MHAD with the proposed cross-subject protocol. Without GCNs and all perform better than the default skeleton-only approach using a single-stream AGCN. Additionally, another benchmark using GCNs on UTD-MHAD does not exist, thus, making a direct comparison of different approaches difficult. From the listing in Table 4.1, it is clear that all fusion approaches skeleton and IMU modalities achieve the highest classification performance out of all methods introduced in this work. In comparison to the best performing fusion approach of skeleton with IMU nodes appended at its central node. MCRL [LKJ19] uses a fusion of skeleton, depth and RGB to reach 93.02% (-1.4%) validation accuracy on UTD-MHAD. Gimme Signals [MTP20a] reach 93.33% (-1.09%) using a CNN and augmented image representations of skeleton sequences. PoseMap [LY18] achieves 94.5% (+0.08%) accuracy using pose heatmaps generated from RGB videos. This method slightly outperforms the proposed fusion approach.

MMAcT To show better generalization, we also conducted experiments on the large-scale MMAcT dataset which contains more modalities, classes and samples as the UTD-MHAD dataset. Note, we only use the cross-subject protocol, the signal modalities can not be separated by view. A comparison of approaches regarding the MMAcT dataset is given in Table 4.2. Kong et al. [Kon+19] propose along with the MMAcT dataset the MMAD approach, a multimodal distillation method utilizing an attention mechanism

Table 4.1: Comparison to the State-of-the-Art on the UTD-MHAD dataset.

Approach	Acc
Skeleton	92.32
RGB Patch Features R-18	27.67
RGB Encoder R-18	27.21
R(2+1)D	61.63
Skeleton + RGB Encoder R(2+1)D	91.62
Skeleton + RGB Encoder R-18	89.83
Skeleton + RGB Patch Features R-18	73.49
Skeleton + RGB Patch Features R-18 (no MLP)	44.60
Skeleton + IMU (Center)	94.42
Skeleton + IMU (Wrist/Hip)	94.07
Skeleton + IMU (Center + Add. Edges)	93.26
Skeleton + IMU (Wrist/Hip + Add. Edges)	93.26
Skeleton + IMU (Channel Fusion)	90.29
Skeleton + IMU + RGB Patch Features R-18	78.90
Skeleton + IMU + RGB Encoder R-18	92.33
Skeleton + IMU + RGB Encoder R(2+1)D	92.85
PoseMap [LY18]	94.50
Gimme Signals [MTP20a]	93.33
MCRL [LKJ19]	93.02

that incorporates acceleration, gyroscope, orientation and RGB. For evaluation, they use the F1-measure and reach an average of 66.45%. Without the attention mechanism, the approach (MMD) yields 64.33%. An approach utilizing the standard distillation approach Single Modality Distillation (SMD) yields 63.89%. The current baseline is set by SAKDN [Liu+21b] which distills sensor information to enhance action recognition for the vision modality. Experiments show that the skeleton-based approach can be further improved by fusion with just the acceleration data to reach a recognition F1-measure of 89.60% (+12.37%). The MMAAct dataset contains two accelerometers, where only the one from the smartwatch yields a mention-able improvement. The most significant improvement of our proposed approach is yielded by introducing the skeleton graph. In contrast, while the fusion approaches of skeleton and all four sensors do not improve the purely skeleton-based approach of 88.65% (+13.41%), with 85.5%

Table 4.2: Comparison to the State-of-the-Art on the MMAct dataset.

Approach	Acc	F1-Measure
Skl	87.85	88.65
Skl+Acc(W+P)+Gyo+Ori	84.85	85.50
Skl+Acc(W+P)+Gyo+Ori (Add. Edges)	84.40	84.78
Skl+Acc(W)	89.32	89.55
Skl+Acc(P)	87.70	88.72
Skl+Gyo	86.35	87.41
Skl+Ori	87.65	88.64
Skl+Acc(W+P)	89.30	89.60
SMD [HVD15] (Acc+RGB)	-	63.89
MMD [Kon+19] (Acc+Gyo+Ori+RGB)	-	64.33
MMAD [Kon+19] (Acc+Gyo+Ori+RGB)	-	66.45
Multi-GAT [II21]	-	75.24
SAKDN [Liu+21b]	-	77.23

(+10.26%) without additional edges and 84.78% (+9.54%) with additional edges, both reach a higher F1-measure than the baseline but also impact the pure skeleton-based recognition negatively.

4.4.4 Ablation Study

Fusion of Skeleton and RGB Skeletons and RGB videos are combined using the three approaches depicted in Fig. 4.4. Fig. 4.4a shows an approach using RGB patches that are cropped around each skeleton node and passed to a ResNet-18 to compute a feature vector $\mathbf{X}_{\text{RGB}} \in \mathbb{R}^{(M \times 512 \times T \times N)}$ as part of preprocessing. The second approach, shown in Fig. 4.4b, uses ResNet-18 to compute a feature vector for each image. The resulting feature vector is rescaled to the size $C_E \cdot N \cdot M$ and reshaped to be able to be fused with skeleton data. Similarly, in Fig. 4.4c, the third approach uses R(2+1)D. In terms of parameters, the basis Skeleton model has 3.454.099 parameters, only 2.532 parameters are added for incorporation of inertial measurements into the model Skeleton+IMU(Center) 3.456.631 for a 2.2% accuracy improvement. Fusion with an RGB encoder adds five times more parameters (Skeleton+RGB Encoder ResNet-18 with 17.868.514) and a massive training overhead.

Table 4.1 shows that the RGB approaches viewed individually (without fusion) do not reach the performance of R(2+1)D pre-trained for action recognition. Results re-

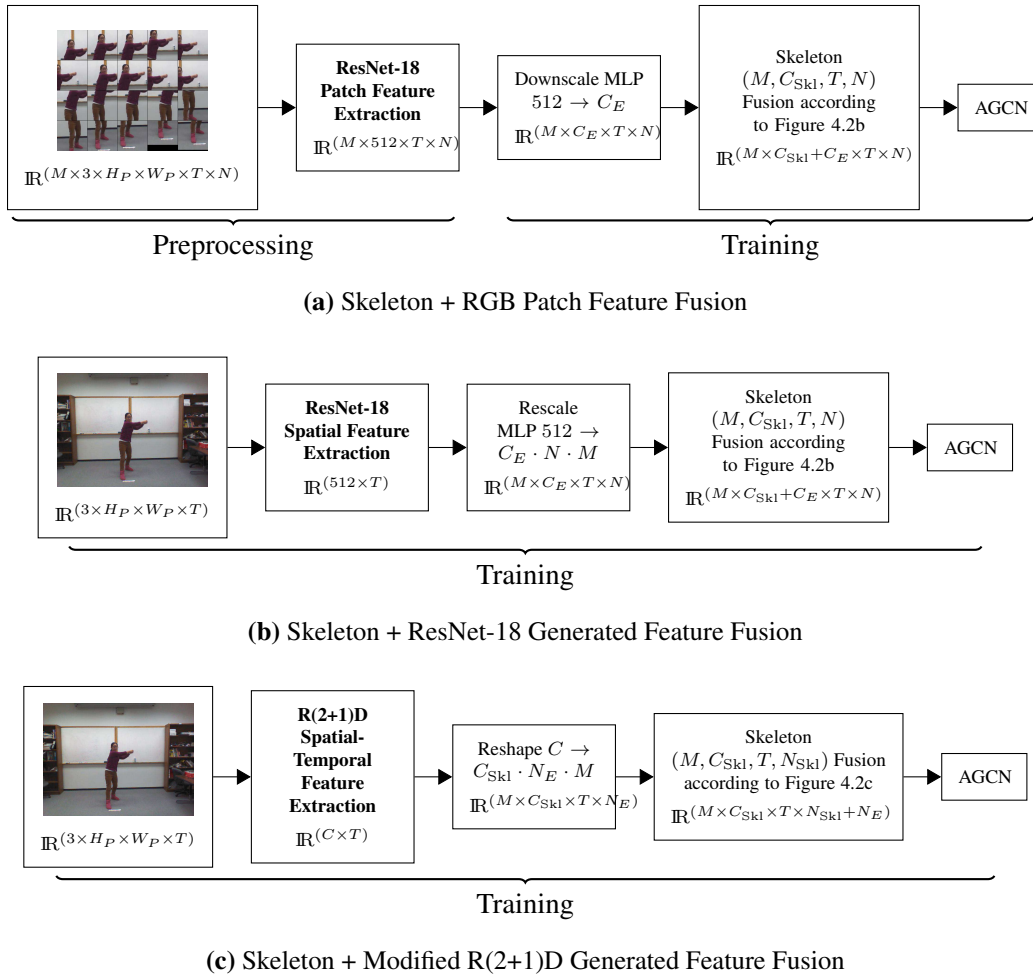


Figure 4.4: The three different skeleton + RGB Fusion models with reference of an image from UTD-MHAD. The first model generates a feature for each node, while the last two generate a feature for the entire image that is distributed to the nodes and adjusted as part of the supervised training.

garding the fusion models show a low accuracy of 73.49% for RGB patch features that have been created outside the training process and 44.6% for the same procedure without a downscaling Multilayer Perceptron (MLP). A similar conclusion can be drawn from the remaining two fusion models. Using R(2+1)D to produce features shows a slightly increased effectiveness of +1.79% (91.62%) over ResNet-18 (89.83%) but -0.7% in comparison to the solely skeleton-based approach.

Fusion of Skeleton and IMU Fusion of skeletal and inertial sensor data is done according to Fig. 4.2. Fig. 4.1 shows the skeleton structure of UTD-MHAD and illustrates

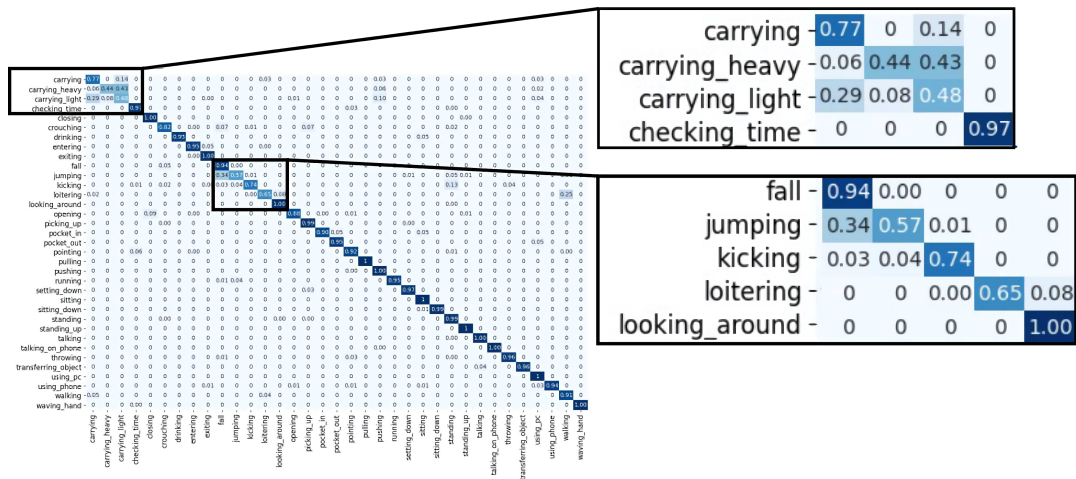


Figure 4.5: Confusion matrix for the results on MMAct with the fusion of skeleton and accelerometer measurements from the smartwatch with highlighted high-confused actions.

two possibilities for fusing the red IMU graph nodes with the skeleton by connecting them to different skeleton joint nodes. In Figure 4.2b, nodes are appended at the central skeleton joint as it is defined in ST-GCN and AGCN papers. The configuration depicted in Figure 4.2b is attributed to the way sensors are worn by subjects of the UTD-MHAD dataset. This configuration is therefore not used for MMAct. Additional configurations arise when additional edges are drawn between the newly added nodes. According to Figure 4.2b, another experiment involves broadcasting the \mathbb{R}^6 -sized IMU feature vector to each skeleton joint and fuse them at channel dimension. From the results in Table 4.1, it is observable that all skeleton graphs with additional associated IMU nodes at each point in time improve the classification performance by at least one percent. In comparison to a skeleton-only approach, variants with additional edges between the newly added nodes perform generally worse than their not-connected counterparts and are both at 93.26% (+0.94%). The average classification accuracy of both other variants reaches 94.42% (+2.1%) and 94.07% (+1.75%). Despite having a slightly increased accuracy for appending new nodes to the existing central node, both variants almost reach equal performance and the location where nodes are appended seemingly does not matter much. While all experiments with fusion at spatial dimension show increased accuracies, the only experiment that does not surpass the skeleton-based approach is about fusion of both modalities at channel dimension, reaching 90.29% (-2.03%) accuracy.

For MMAct, all experiments are conducted using only the configuration in Fig. 4.1a and its variation with interconnected nodes. Table 4.2 shows that the skeleton-based approach reaches 87.85% accuracy for a cross-subject split, fusion approaches including all four sensors perform worse and reach only 84.85% (-3%) and 84.4% (-3.45%). Mixed results are achieved when individual sensors are not part of the fusion model. Fu-

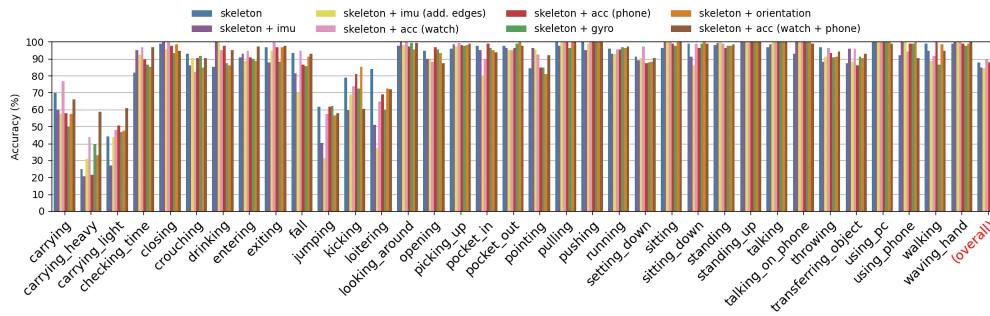


Figure 4.6: Class specific accuracy for all MMAct classes for the fusion of various data modalities with Fusion-GCN.

Class	Skl	Skl + Acc
carrying_heavy	24.69	43.83
checking_time	81.93	96.58
drinking	85.00	95.00
transferring_object	87.23	96.10
pointing	84.52	92.34

Table 4.3: Top-5 most improved classes by the fusion of skeleton (Skl) and additional accelerometer (Acc) data from the smartwatch.

sion using only one of the phone’s individual sensors, acceleration, gyroscope or orientation, reaches comparable results with 87.70% (-0.15%), 86.35% (-1.5%) and 87.65% (-0.2%) accuracy, respectively. On the contrary, performing a fusion of skeleton and acceleration data obtained by the smartwatch or with the fusion of both acceleration sensors shows an improved accuracy of 89.32% (1.47%) and 89.30% (1.45%).

Table 4.3 shows the top-5 improved classes by the fusion with the accelerometer measurements of a smartwatch. All the top-5 improved actions have a high arm movement in common. In Fig. 4.5 we give a confusion matrix for the Skeleton + Accelerometer (Watch) and highlight the most confused classes. Especially the variations of the “carrying” actions are hard to distinguish by their obvious similarity. Also, actions that contain sudden movements with high acceleration peaks are often confused (“jumping” is often considered as “falling”). This might be caused by the extraction of the human poses with a lower frame rate where an important part is then missing. For instance if the first part of the jumping activity is missing then the human poses might just contain the part where the person is landing which could lead to a misclassification as falling. In general, most of the activities can be recognized quite well. Fig. 4.6 gives a general comparison of all class-specific results on different fusion experiments. Especially the fusion from skeleton sequences with the accelerometer measurements (skeleton + acc

(watch)) suggest a high improvement on many classes, where the similar “carrying” classes are to highlight.

Fusion of Skeleton, RGB and IMU One experiment is conducted using skeleton, RGB and IMU with IMU nodes appended to the skeleton central node without additional edges in combination with and all three RGB early fusion approaches. The results in Table 4.1 show that, like previously except for the RGB patch feature model, all models achieve an accuracy over 90%, albeit not reaching the same values as the skeleton and IMU fusion approach.

4.4.5 Limitations and Discussion

Comparing skeleton and skeleton + IMU, the fused approach generally has less misclassifications in all areas. Especially similar actions, such as “throw”, “catch”, “knock” or “tennis swing”, are able to be classified more confidently. The only action with decreased recognition accuracy using the fused approach is “jog” which is misclassified more often as “walk”, two similar actions and some of the few with sparse involvement of arm movement. Common problems for all RGB approaches regarding UTD-MHAD are a small number of training samples, resulting in overfitting in some cases that can not be lifted by either weight decay or dropout. Another fact is the absence of object interactions in UTD-MHAD. With the exception of “sit2stand” and “stand2sit”, actions such as “throwing”, “catching”, “pickup_throw” or sports activities never include any objects. As pointed out previously, skeleton is focused purely on human movements and, by that, omits all other objects inside of a scene. RGB still contains such visual information, making it supposedly more efficient in recognizing object interactions. In contrast, many of MMAAct’s actions, like “transferring_object”, “using_pc”, “using_phone” or “carrying”, make use of real objects. While fusion with RGB modality achieves similar accuracies as other approaches, incorporating the data into the network increases the training time by up to a magnitude of ten; hence, the RGB fusion models do not provide a viable alternative to skeleton and IMU regarding the current preprocessing and training configurations. Therefore, due to timely constraints, experiments for fusion of skeleton and RGB modalities on the larger dataset MMAAct are omitted.

4.5 Conclusion

With *Fusion-GCN*, we presented an approach for multimodal action recognition using GCNs. To incorporate additional modalities, we propose to fuse on two different dimensions, either on a channel or spatial dimension. Further integration into early and late fusion approaches have been presented. In our experiments, we considered the

flexible fusion of skeleton sequences with inertial measurements, accelerometer-, gyro-, orientation- measurements separately, as well as RGB features. Our presented fusion approach successfully improved the previous baselines on the large-scale MMAct dataset significantly. A large improvement is based on the usage of skeleton sequences in conjunction with a GCN-based model with additional improvements attributing to the fusion with additional modalities, where especially the fusion with the smartwatch data has been shown to improve the action recognition performance. However, adding to many modalities led too uncertainty and decreased the performance. We believe that *Fusion-GCN* demonstrated successfully that GCNs serve as good basis for multimodal action recognition and could potentially guide future research in this domain.

Chapter 5

One-Shot Action Recognition

In this chapter, we present approaches for the action recognition task in a semi-supervised setting. Similar to approaches presented in Chapter 3, we represent various sensor-modalities in an image. Models are trained on a set of known action classes and tested on a distinct set of unknown classes. When referring to one-shot recognition [Liu+20a], one sample per unseen class is provided in a reference set.

This chapter is based on our previous publications [MTP20b; Mem+22]. In the first part of this chapter, we focus on a signal-level problem formulation with the goal to propose a one-shot action recognition approach that is applicable for various sensor modalities. With our Signal-Level Deep Metric Learning (*SL-DML*) [MTP20b] approach, we propose a metric learning approach to reduce the action recognition problem to a nearest neighbor search in embedding space. We encode signals into images and extract features using a deep residual CNN. Using triplet loss, we train an embedding function. The resulting encoder transforms features into an embedding space where closer distances encode similar actions while higher distances encode different actions. Our approach is based on a signal-level formulation and remains flexible across a variety of modalities.

In the second part of this chapter, we concentrate on skeleton-based one-shot action recognition. With our Skeleton-Based Deep Metric Learning (*Skeleton-DML*) [Mem+22] approach, we follow the idea of *SL-DML*, but focus on the one-shot action recognition of skeleton sequences. A novel representation is proposed and compared against a wide set of comparable skeleton-sequence representations known from supervised action recognition tasks.

5.1 Introduction

Learning to identify unseen classes from a few samples is an active research topic. Metric learning in computer vision research mainly concentrates on one-shot object recog-

nition [Fu+15], person re-identification [Yi+14; WB18] or face identification [SKP15]. Only recently, few-shot methods for action recognition [Car+19; JM19; Liu+20a; Pen+21] have gained popularity. These approaches present good results for one-shot action recognition, but only concentrate on single modalities like image- or skeleton sequences. We propose to use a signal level representation that allows flexible encoding of signals into an image and the fusion of signals from different sensor modalities.

In contrast to classification methods, which predict class labels, metric learning approaches learn an embedding function. Our approaches learn a function that embeds signal- or skeleton-representations into an embedding space. One-shot action recognition then becomes a nearest neighbor search in embedding space. Figure 5.1 gives an application example for one-shot action recognition on skeleton sequences using our approach.

The first part of the chapter concentrates on a signal-level formulation of the one-shot action recognition problem. While it may appear implausible, initially, to encode signals into an image representation for action recognition, it entails some benefits. First, it allows generalization across different sensor modalities as long as a sensor originates multivariate signal sequences or higher-level features such as human pose estimates. There is no need for modality-specific architectures or pipelines. Further, an image-like representation allows the usage of well-studied and well-performing classification architectures [He+16]. Finally, experiments for multi-modal or inter-modal one-shot action recognition can be conducted flexibly.

In our study, signals originate from 3D skeleton sequences gathered by an RGB-D camera, inertial, or motion capturing measurements. To fuse multiple modalities, e.g., skeleton sequences and inertial measurements, the signal matrices are concatenated and represented as an image. Inter-modal experiments are especially interesting, as they allow training on one modality and recognition on another, previously unseen, modality by providing only a single reference. A new sensor can be used for action recognition without any prior training data from that sensor.

In the second part of the chapter, we focus on skeleton-based one-shot action recognition. RGB-D cameras that support the OpenNI SDK not only provide color and depth streams, but also offer human pose estimates in the form of skeleton sequences. These skeleton estimates allow a wide variety of higher-level applications without investing in the human pose estimation problem. As the pose-estimation approach is based on depth streams [Zha12], it is robust against background information and different lighting conditions. Therefore, it also remains functional in dark environments. Especially in a robotics context, one-shot action recognition enables a considerable variety of applications to improve the human-robot-interaction. A robot could initiate a dialog, when recognizing an activity that it is unfamiliar with, to assign a robot-behavior to the observation. This can be done with a single reference sample, while standard action recognition approaches can only recognize actions that were given during training time. In

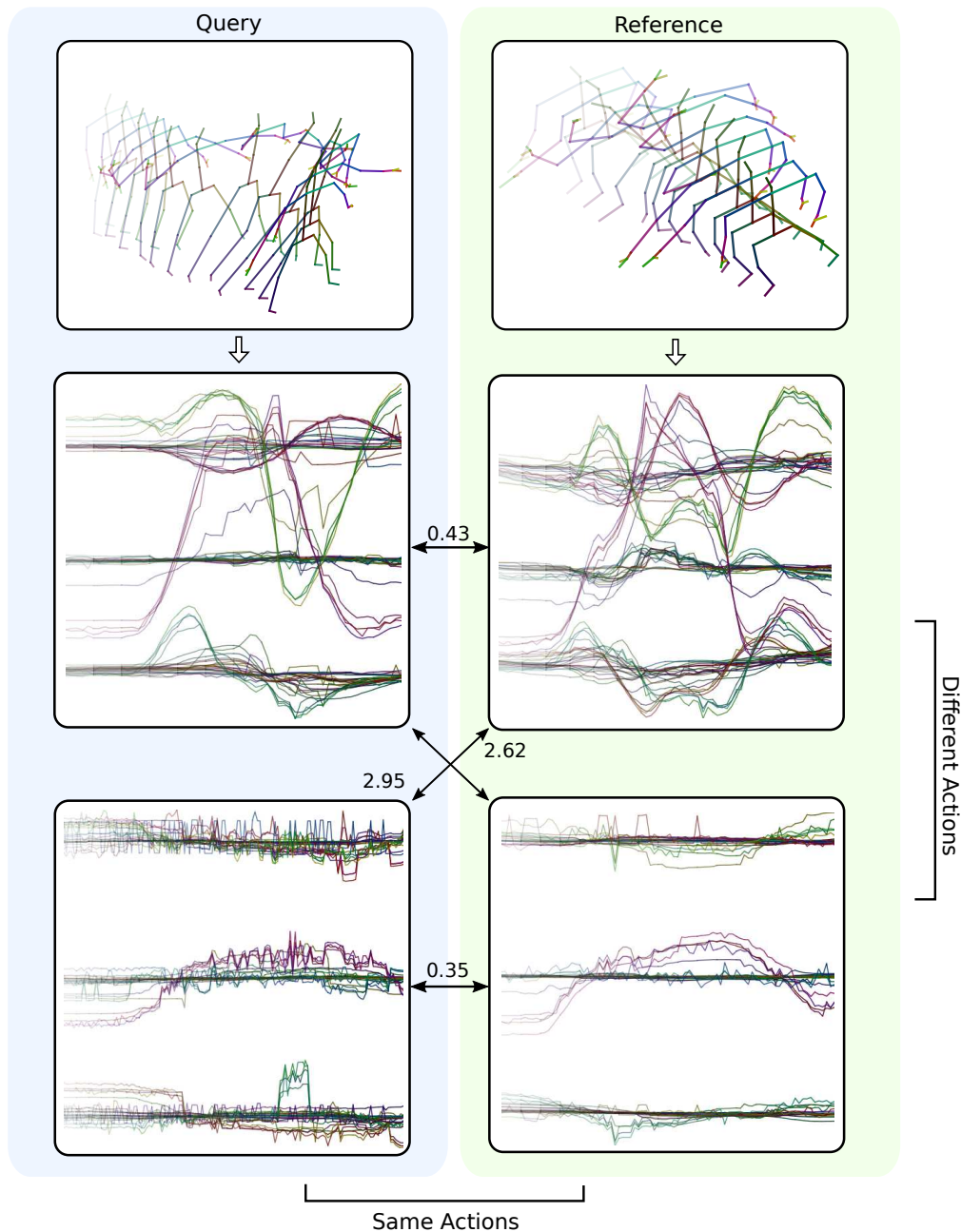


Figure 5.1: Illustrative example. In this example, a skeleton is transformed into an image-like representation. Joint axes are encoded as signals. Each axis is drawn in a different color. Our approach encodes an action sequence representation into an embedding vector. Low Euclidean distances on the action embedding represent close similarity, whereas higher distances represent different actions. This approach allows for one-shot action classification or clustering of similar activities. The underlying signal level representation enables multi-modal applications.

our proposed one-shot action recognition approach, observations are projected to an embedding space in which similar actions have a low distance, and dissimilar actions have a high distance. A high distance to all known activities can be seen as an indicator for anomalies. The embedding in a metric learning setting allows online association of novel observations, which is a high advantage over classification tasks that would require retraining or fine-tuning.

The contributions of this chapter are as follows:

- We present a novel model for one-shot action recognition on a signal level.
- We present a representation that reassembles skeleton sequences into images.
- We integrate the representation into a deep metric learning formulation to tackle the one-shot action recognition problem.
- We furthermore evaluate related skeleton-based image representations for one-shot action recognition.

A classifier and embedding encoder are jointly optimized using *triplet margin loss* [WS09] with a *Muti-Similarity Miner* [Wan+19b]. The nearest neighbor in embedding space defines the most similar action. Our proposed approaches lift the *state-of-the-art* in one-shot action recognition on skeleton sequences on the NTU RGB+D 120 dataset for the one-shot evaluation protocol by 5.6% and with *Skeleton-DML* by an additional +3.3%, while *SL-DML* still generalizes for other sensor data modalities like IMU and motion capturing systems. For our *SL-DML* approach, we claim that our approach based on triplet margin loss and a common signal-level representation yields high flexibility for applications in one-shot action recognition. We achieve good results on one-shot action recognition for conventional sensor modalities (skeleton sequences, inertial measurements, motion capturing measurements). Our approach shows good capabilities when being trained on one modality and inferred on a different modality by providing a single reference sample per action class of the unknown modality. This allows. e.g., training on skeleton sequences and inference on inertial measurements. With *Skeleton-DML*, we further improve on the initial idea from *SL-DML* but focus on skeleton-based action recognition. A novel image-based representation is proposed and compared against other related representations in the context of one-shot action recognition. Further, we use the Multi-Similarity-Loss for our *Skeleton-DML* approach.

5.2 Related Work

We give a brief overview of methods related to metric learning and few-shot recognition approaches in general. We focus on methods for action embeddings and few-shot action recognition.

Action recognition is a broad research topic that varies not only in different modalities like image sequences, skeleton sequences, data by inertial measurement units but also by their evaluation protocols. Most common protocols are cross-view or cross-subject. More recently, one-shot protocols have gained attention. As our approach focuses on skeleton-based one-shot action recognition, we present related work from the current research state directly related to our method. Skeleton-based action recognition gained attention with the release of the Microsoft Kinect RGB-D camera. This RGB-D camera not only streams depth and color images, but the SDK also streams skeleton data processed from the depth images. With the *NTU RGB+D* dataset [Sha+16; Liu+20a] a large-scale RGB-D action recognition dataset that also contains skeleton sequences has been released. The progress made on this dataset gives a good indication of the performance of various skeleton-based action recognition approaches. The idea of representing motion in image-like representations lead to serious alternatives to sequence classification approaches based on *Recurrent Neural Networks* [Hu+19] and *Long Short Term Memory (LSTM)* [Liu+18a].

Metric Learning Metric learning has been intensively studied in computer vision. A focus is on metric learning from photos or cropped detection boxes for person re-identification or image-ranking. Schroff et al. [SKP15] presented a joint face recognition and clustering approach. They trained a network such that the squared L2 distances in the embedding space directly correspond to face similarity [SKP15]. Triplet loss [WS09] is used for training the embedder. The embedding minimizes distances between anchor images and positive images (i.e., same person, different viewpoint) and maximizes distances to negative samples (different person). Yi et al. [Yi+14] presented a deep metric learning approach based on a Siamese deep neural network for person re-identification. The two sub-nets are combined using a cosine layer. Wojke and Bewley [WB18] propose a deep cosine metric learning approach for the person re-identification task. The *Cosine Softmax Classifier* pushes class samples towards a defined class mean and therefore allows similarity estimation by a nearest neighbor search.

Skeleton Representations Because convolution neural architectures showed great performance in the image-classification domain, a variety of research concentrated on finding image-like representations for different research areas like speech recognition [Her+17]. Our one-shot action recognition approaches build on image representation of signal sequences. Prior work has already presented representations for action recognition with skeleton sequences. The *Skeleton-DML* approach, presented later, is based on representations for skeleton sequences. Thus, we now introduce some representation approaches for skeleton-based action recognition. We use these representations in our experiments for the *Skeleton-DML* approach. Representations for encoding spatio-temporal information were explored in-depth for recognizing actions [LLC17; Cae+19].

They focus on a classification context by associating class labels with skeleton sequences, in contrast to learning an embedding space. Caetano et al. [Cae+19; CBS19] represent a combination of reference joints and a tree-structured skeleton as images. Their approach preserves spatio-temporal relations and joint relevance. In contrast to our approach, their underlying representation enforces custom network architectures and is constrained to skeleton sequences, whereas our *SL-DML* approach adds flexibility to other sensor modalities. Liu et al. [LLC17] presented a combination of skeleton visualization methods and jointly trained them on multiple streams. Wang et al. [Wan+18a] presented joint trajectory maps. Viewpoints from each axis were set and encoded 3D trajectories for each of the three main axis views. A simple Convolutional Neural Network (CNN) architecture was used to train a classifier analyzing the joint trajectory maps. Occlusion could not be directly tackled. Therefore, the representation by Liu et al. [LLC17] added flexibility by fusing up to nine representation schemes in separate image channels. A similar representation has recently shown to be usable also for action recognition on different modalities and their fusion [MTP20a]. Kim and Reiter [KR17], on the other hand, presented a compact and human-interpretable representation. Joint movement contributions over time can be interpreted. Interesting to note is the skeleton transformer by Li et al. [Li+17]. They employ a fully connected layer to transform skeleton sequences into a 2 dimensional matrix representation. Yang et al. [Yan+19] present a joint order that puts joints closer together if their respective body parts are connected. It is generated by a depth-first tree traversal of the skeleton starting in the lower chest. Skepxels are small 5×5 -pixel segments containing the positions of all 25 skeleton joints in a random but fixed order. Liu et al. [LAM19] use this 2D structure as it is more easily captured by CNNs. Each sample of a sequence is turned into multiple sufficiently different Skepxels, which are then stacked on top of each other. These Skepxels differ only in their joint permutation. The full Skepxel image of a skeleton's sequence is assembled width-wise, without altering the joint permutation within one row of Skepxels. Caetano et al. [Cae+19] generate two images containing motion information in the form of an orientation and a magnitude. The orientation is defined by the angles between the motion vector and the coordinate axes. The angles are stored in the color channels of an image with time in horizontal and the joints in TSSI order in vertical direction. Instead, the gray-scale magnitude image contains the Euclidean norm of the motion vectors.

Action Embedding A recent action embedding approach by Hahn et al. [HSR19] takes inspiration from the success of word embeddings in natural language processing. They combine linguistic cues from class labels with spatio-temporal features from sequences. A hierarchical recurrent neural network trains a feature extractor. A joint loss combines classification accuracy and similarity trains a function to encode the input into an embedding. Discriminative embeddings are important for few-shot learning ap-

proaches. Jasani and Mazagonwalla [JM19] proposed a similar approach for skeleton-based zero-shot action recognition. A *Spatio Temporal Graph Convolution Network (ST-GCN)* [YXL18] extracts features which are encoded in semantic space by a continuous bag of words method.

One-Shot Action Recognition One-shot recognition in general aims at finding a method to classify new instances with a single reference sample. Possible approaches for solving problems of this category are metric learning [Wan+14; HA15], or meta-learning [FAL17]. In action recognition, this means a novel action can be learned with a single reference demonstration of the action. Contrary to one-shot image classification, actions consist of sequential data. A single frame might not contain enough context to recognize a novel activity. One-shot action recognition is in comparison to image ranking, or person re-identification a quite underrepresented research domain. Kliper-Gross et al. [KHW11] proposed One-Shot-Similarity Metric Learning. A projection matrix that improves the One-Shot-Similarity relation between the example same and not-same training pairs represents a reduced feature space [KHW11]. Fanello et al. [Fan+13] use *Histogram of Flow* and *Global Histogram of Oriented Gradient* descriptors with adaptive sparse coding and are classified using a linear SVM. Careaga et al. [Car+19] propose a two-stream model for few-shot action recognition on image sequences. They aggregate features from optical flow and the image sequences separately by a LSTM and fuse them afterward for learning metrics. Rodriguez et al. [Rod+17] presented a one-shot approach based on *Simplex Hidden Markov Models (SHMM)*. Improved dense trajectories are used as base features [WS13]. A maximum a posteriori (MAP) adoption and an optimized Expectation Maximization reduce the feature space. A maximum likelihood classification, with the SHMM, allows one-shot classification. Roy et al. [RMM18] propose a Siamese network approach for discriminating actions by a contrastive loss on a low dimensional representation gathered factory analysis. Mishra et al. [Mis+18] presented a generative framework for zero- and few-shot action recognition on image sequences. A probability distribution models classes of actions. The parameters are functions for semantic attribute vectors that represent the action classes. Along with the *NTU RGB+D 120* dataset, Liu et al. [Liu+20a] presented a one-shot action recognition protocol and corresponding baseline approaches. The *Advanced Parts Semantic Relevance (APSR)* approach extracts features by using a spatio-temporal LSTM method. They propose a semantic relevance measurement similar to word embeddings. Body parts are associated with an embedding vector and a cosine similarity is used to calculate a semantic relevance score. Sabater et al. [Sab+21] presented a one-shot action recognition approach based on a Temporal Convolutional Network (TCN). After normalization of the skeleton stream, they calculate pose features and use the TCN for the generation of motion descriptors. The descriptors at the last frame, assumed to contain all relevant motion from the skeleton-sequence, are used to calculate the distances to the

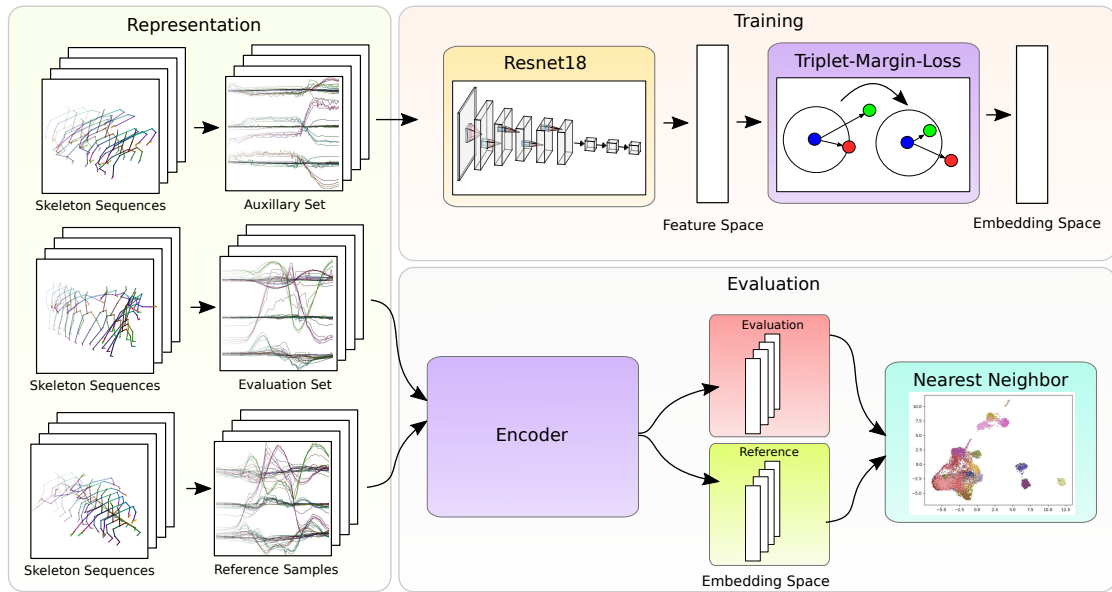


Figure 5.2: Approach overview: We represent actions on a signal level. In the example above, we transformed skeleton joint axes into images. We use a ResNet18 architecture with a triplet loss to train a model that transforms an image into an embedding space. For inference, the trained encoder encodes a set of references and queries. The closest reference in embedding space represents the most similar activities for which we use a nearest-neighbor search.

reference samples. Action classes are associated by thresholding the distances.

Multi-modal Few-Shot Action Recognition The field of multi-modal few-shot action recognition is entirely unexplored. Somehow related is the work of Al-Naser et al. [Al+18], who presented a zero-shot action recognition approach by combining gaze guided object recognition with a gesture recognition wrist-band. Actions are detected by fusing features of sub-networks per modality and integrating action definitions. Only three actions demonstrate the recognition results. Very recently, a multi-modal fusion transformer has been presented by Shvetsova et al. [Shv+21]. This approach operates on video, audio and text-input that is represented in a joined multi-modal representation. The model is trained with a combinatorial loss that projects each of the modalities into a joint embedding space. The method has shown to perform well for the zero-shot action localization task.

5.3 Signal-Level Deep Metric Learning

To cover the action recognition task across a variety of sensor modalities, we consider the action recognition problem on a signal level. Signals are encoded in a discriminable

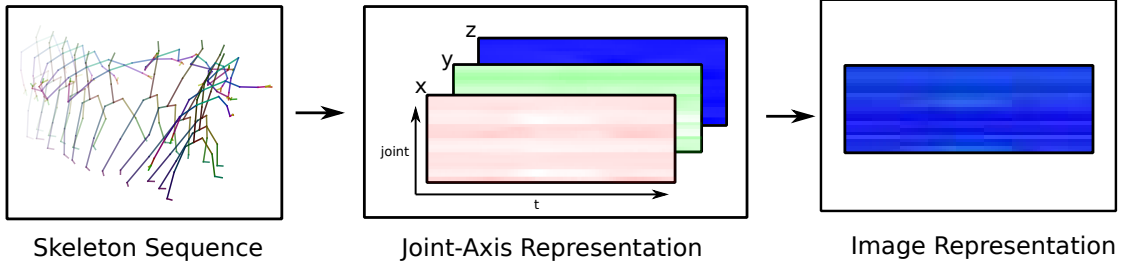


Figure 5.3: Exemplary representation for a throwing activity of the NTU-RGB+D 120 dataset.

image representation. An image-like representation allows direct adaption of already established image classification architectures for extracting features. On the extracted features, we train a similarity function yielding an action embedding function using triplet loss. The triplet loss minimizes embedding distances between similar action samples while maximizing distances between different actions. Finally, to solve the one-shot problem, we apply a nearest neighbor search in the embedding space. An illustration of our approach is given in Fig. 5.2.

5.3.1 Problem Formulation

The one-shot action recognition problem is considered as a metric learning problem. First, we encode action sequences on a signal level into an image representation. The input in our case is a signal matrix $\mathbf{S} \in \mathbb{R}^{N \times M}$ where each row vector represents a discrete 1-dimensional signal and each column vector represents a sample of all sensors at one specific time step. The matrix is transformed to an RGB image $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$ by normalizing the signal length M to W and the range of the signals to H . The identity of each signal is encoded in the color channel. This results in a dataset $\mathcal{D} = \{(\mathbf{I}_i, \mathbf{y}_i)\}_{i=1}^K$ of K training images $\mathbf{I}_{1, \dots, K}$ with labels $\mathbf{y}_i \in \{1, \dots, C\}$. Our goal is to train a feature embedding $\mathbf{x} = g_{\Theta}(\mathbf{I})$ with parameters Θ which projects input images $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$ into a feature representation $\mathbf{x} \in \mathbb{X}^d$. The feature representation reflects minimal distances for *similar* classes.

5.3.2 Representations

Our approach builds upon a discriminable image representation. Therefore, we propose a novel, more compact signal level representation. Multivariate signal or higher-level feature sequences are reassembled into a 3 channel image. Each row of the resulting image corresponds to one joint, and each channel corresponds to one sample in the sequence. The color channels, red, green and blue, represent respectively the signals' x-, y- and z-values. The resulting images are normalized to the range of 0 to 1. We chose to normalize over the whole image to preserve the relative magnitude of the signals.

In contrast to the sparse representations used for multi-modal action classification in Chapter 3 or skeleton-based action recognition [Wan+18a; LLC17], the proposed representation is invertible and more compact. This representation conforms the dense representation as used in Chapter 3. Chronological the representation has first been used for the one-shot action recognition experiments and afterwards extended our action recognition experiments. The construction of the representation is depicted in Fig. 5.6.

5.3.3 Feature Extraction

Most action recognition approaches based on CNNs present custom architecture designs in their pipelines [LLC17]. A benefit is the direct control over the number of model parameters that can be specifically engineered for data representations or use cases. Recent advances in architecture design cannot be transferred directly. Searching good hyperparameters for training is then often an empirical study. Minor architecture changes can result in an entirely different set of hyperparameters. He et al. [He+16] suggested the use of residual layers during training to tackle the vanishing gradient problem. We take advantage of the recent development in architecture design and decided to use a ResNet18 [He+16] architecture. For weight initialization, we use a pre-trained model. After the last feature layer, a two-layer perceptron to transform the features into the embedding size is applied. The embedding is refined by the metric learning approach.

5.3.4 Metric Learning

Metric learning aims to learn a function to transform an image into an embedding space, where the embedding vectors of similar samples are encouraged to be closer, while dissimilar ones are pushed apart from each other [Wan+19b]. We use a triplet loss with a *Multi-Similarity-Miner* [Wan+19b] for mining good triplet candidates during training.

While the triplet loss has been used in image ranking [Bui+17], face recognition [SKP15], and person re-identification [HBL17] it has only rarely been used for inter- and cross-modal ranking to improve action recognition [Wan+18b] or for complex event detection [Hou+18]. Given a triplet of an anchor image I_o , a positive data sample, representing the same action class image I_\uparrow and a negative sample, representing a different action class I_\downarrow the triplet loss can be formulated as:

$$\mathcal{L}_{\text{triplet}}(I_o, I_\uparrow, I_\downarrow) = \max(\|g(I_o) - g(I_\uparrow)\|_2 - \|g(I_o) - g(I_\downarrow)\|_2 + \delta, 0),$$

where δ describes an additional distance margin.

Finding good candidate pairs is crucial. Therefore, we use a *Multi-Similarity Miner* [Wan+19b] to mine positive and negative pairs that are assumed to be difficult to push

apart in the embedding space. That means positive pairs are constructed by an anchor, its positive image pair $\{\mathbf{I}_o, \mathbf{I}_\uparrow\}$ and its embedding $g(\mathbf{I}_o)$, preferring pairs with a high distance in embedding space with the following condition:

$$\|g(\mathbf{I}_o) - g(\mathbf{I}_\uparrow)\|_2 > \min_{\mathbf{y}_k \neq \mathbf{y}_i} \|g(\mathbf{I}_i) - g(\mathbf{I}_k)\|_2 - \epsilon,$$

likewise, negative pairs $\{\mathbf{I}_o, \mathbf{I}_\downarrow\}$ are mined by the lowest distance in embedding space:

$$\|g(\mathbf{I}_o) - g(\mathbf{I}_\downarrow)\|_2 < \max_{\mathbf{y}_k \neq \mathbf{y}_i} \|g(\mathbf{I}_i) - g(\mathbf{I}_k)\|_2 + \epsilon,$$

where ϵ is a given margin. Finally, we yield the total loss by:

$$\mathcal{L} = \alpha \mathcal{L}_{\text{triplet}} + \beta \mathcal{L}_{\text{classifier}},$$

such that the influences of the loss can be weighted using the scalars α for the triplet loss $\mathcal{L}_{\text{triplet}}$ and β for the classifier loss $\mathcal{L}_{\text{classifier}}$. We utilize a cross entropy loss for $\mathcal{L}_{\text{classifier}}$. Finding an action class by a query and set of references is now reduced to a nearest-neighbor search in the embedding space. The classifier and encoder are jointly optimized. After the last feature layer of the classifier, a two-layer perceptron is used to yield an embedding size of 128.

5.4 Skeleton-Based Deep Metric Learning

In the previous section (Section 5.3), we presented an one-shot action recognition approach on signal-level. This section sets a focus on the skeleton-based one-shot action recognition. Fig. 5.4 shows an illustrative example of an application of our approach. We propose a novel, compact image representation for skeleton sequences. Additionally, we present an encoder model that learns to project said representations into a metric embedding space that encodes action similarity.

5.4.1 Problem Formulation

The problem formulation for the *Skeleton-DML* approach, except the focus on skeleton sequences, aligned with the problem formulation from Section 5.3.1. A standard approach for action recognition is trained on C classes where the training and test sets share the same C classes. Thus, a test set \mathcal{T} shares the same classes as the training set \mathcal{D} . In an one-shot action recognition setting C classes are known in a auxiliary training set \mathcal{D} , while the test set \mathcal{T} contains U novel classes, providing a single reference sample per class in an reference set \mathcal{A} , where $|\mathcal{A}| = U$. We consider the one-shot action recognition problem as a metric learning problem. Our goal is to train a feature embedding $\mathbf{x} = g_\Theta(\mathbf{I})$ with parameters Θ which projects input images $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$

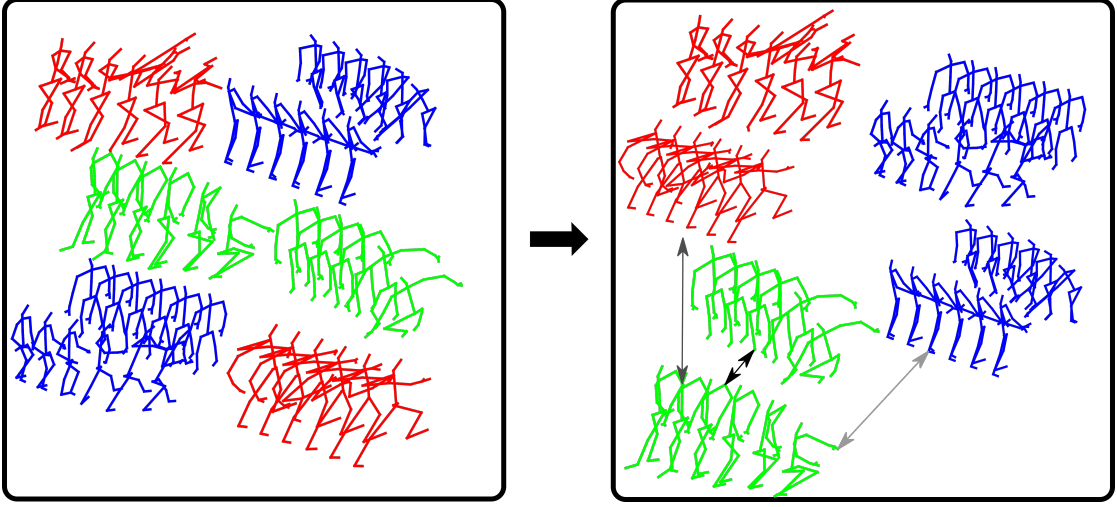


Figure 5.4: Illustrative example of our method. Prior to training a metric on the initial data, no class association could be formed given a skeleton sequence. After training our one-shot action recognition model, skeleton sequences can be encoded. A euclidean distance on the encoded sequence allows class association by finding the nearest neighbour in embedding space from a set of reference samples. The colors are encoding the following classes: **throw**, **falling**, **grab other person’s stuff**. Brighter arrow colors denote higher distance in embedding space.

into a feature representation $\mathbf{x} \in \mathbb{X}^d$. H denotes the height of the image, W denotes the width of the image in an RGB channel image and d is the given target embedding vector size. The feature representation reflects minimal distances in embedding space for *similar* classes. For defining the similarity, we follow [Wan+19b], where the similarity of two samples (I_i, \mathbf{x}_i) and (I_j, \mathbf{x}_j) is defined as $D_{ij} := \langle \mathbf{x}_i, \mathbf{x}_j \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the dot product, resulting in an $K \times K$ similarity matrix \mathbf{D} .

5.4.2 Skeleton-DML Representation

We encode skeleton sequences into an image representation. Fig. 2.9b shows the skeleton as contained in the NTU RGB+D 120 dataset. On a robotic system, these skeletons can be either directly extracted from the RGB-D camera [Zha12], or from a camera image stream using a human-pose estimation approach [Cao+21]. The input in our case is a skeleton sequence matrix $\mathbf{S} \in \mathbb{R}^{N \times M}$ where each row vector represents a discrete joint sequence (for N joints) and each column vector represents a sample of all joint positions at one specific time step of a sequence length M . The matrix is transformed to an RGB image $\mathbf{I} \in \{0, \dots, 255\}^{H \times W \times 3}$. In contrast to [MTP20b; DFW15] the joint space is not projected to the color channels but unfolded per axis separately like depicted in Fig. 5.5, and Fig. 5.6. This results in a dataset $\mathcal{D} = \{(\mathbf{I}_i, \mathbf{y}_i)\}_{i=1}^K$ of K training images

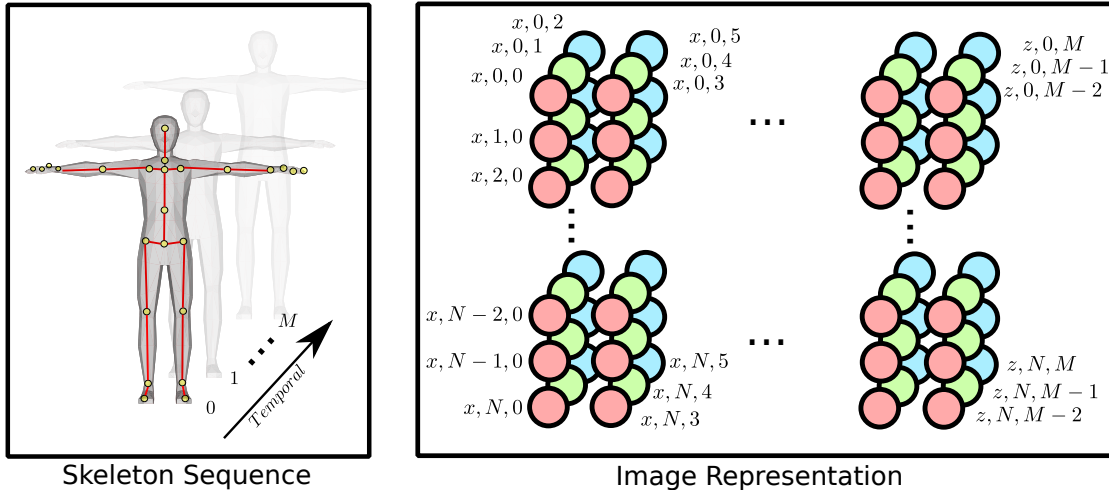


Figure 5.5: *Skeleton-DML* skeleton representation. x and z denote the skeleton joint component in joint space. The number of joints is reflected by N , which relates to the height of the image H . The sequence length M relates to the width of the image W . Instead of projecting the temporal information throughout the width of the image, we project the joint space locally for each dimension and assemble the joint axis blocks over the width.

$I_{1,\dots,K}$ with labels $y_i \in \{1, \dots, C\}$. In contrast to the representations used for multimodal action recognition [MTP20a] or skeleton based action recognition [Wan+18a; LLC17] the proposed representation is more compact. In comparison to [DFW15] and the representation for *SL-DML*, our representation separates the joint values for all axes as blocks over the width, keeping all joint values grouped locally together per axis. In [MTP20b] the color channels are used to unfold the joint values. As the skeleton-sequence is represented as an image, the model needs to be applied only to a single image for inference.

5.4.3 Feature Extraction

For better comparability between the approaches, we use the same feature extraction method as previously proposed in *SL-DML* [MTP20b]. Using a ResNet18 [He+16] architecture allows us to train a model that converges fast and serves as a good feature extractor for the embedder. The low number of parameters allows practical use for inference on autonomous mobile robots. Weights are initialized with a pre-trained model and are optimized throughout the training of the embedder. After the last feature layer, we use a two-layer perceptron to transform the features to the given embedding size. The embedder is refined by the metric learning approach.

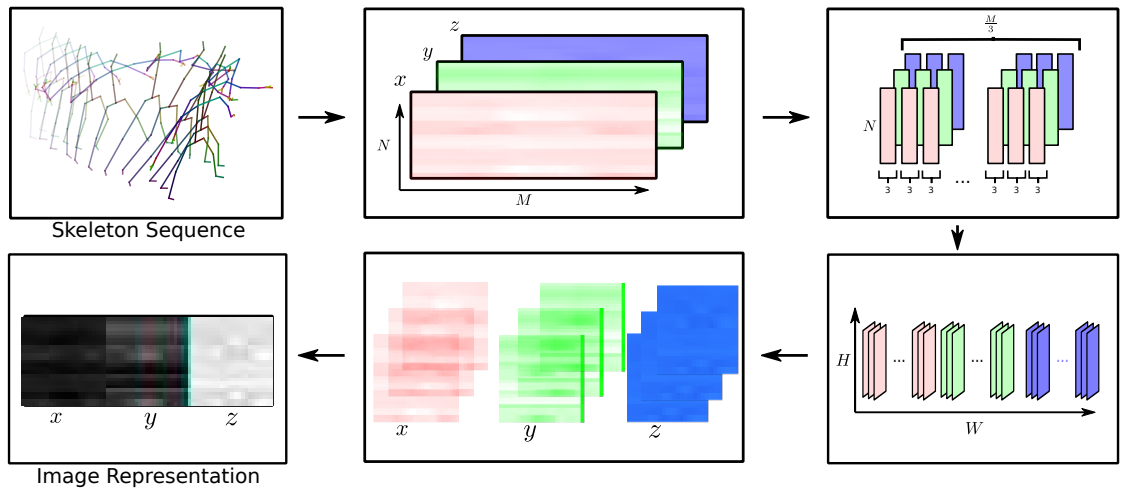


Figure 5.6: Exemplary representation for a throwing activity of the *NTU-RGB+D 120* dataset. A skeleton-sequence serves an input and can be represented as an image directly [DFW15; MTP20a]. Our *Skeleton-DML* representation groups x -, y -, z joint values locally in $\frac{M}{3}$ blocks per axis and assembles them into the final image representation. All axis blocks are laid out aside.

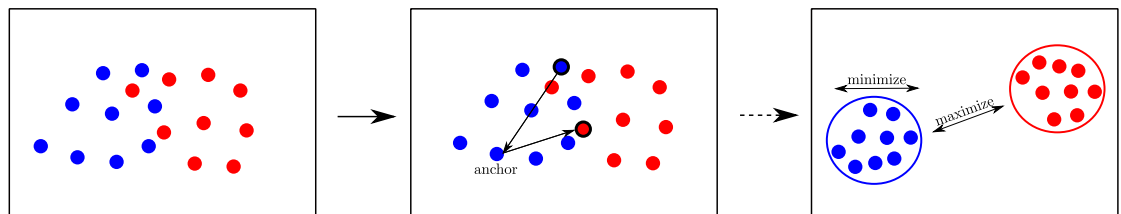


Figure 5.7: A possible intermediate state of the embeddings during the training process of two classes (left). During training, pairs, that are difficult to push apart in embedding space, are mined (middle). Given the blue anchor sample, the most difficult positive pair is the blue sample with the highest distance in embedding space. Similar, the closest red sample in embedding space is the corresponding negative sample. The goal is to separate the samples in embedding space (right) by minimizing the inter-class scatter and maximize the intra-class distance to the class centers in embedding space.

5.4.4 Metric Learning

We aim to learn a function that projects an skeleton image-representation into an embedding space, where the embedding vectors of similar samples are encouraged to be closer, while dissimilar ones are pushed apart from each other [Wan+19b]. We use a *Multi-Similarity-Loss* with a *Multi-Similarity-Miner* [Wan+19b] for mining good pair candidates during training. Positive and negative pairs (by class label), that are assumed to be difficult to push apart in the embedding space, are mined. Fig. 5.7 gives

a constructed example of how positive and negative pairs are mined. Positive pairs are constructed by an anchor and positive image pair $\{\mathbf{I}_o, \mathbf{I}_\uparrow\}$ and its embedding $g(\mathbf{I}_o)$, preferring pairs with a low similarity in embedding space (high distance in embedding space) with the following condition:

$$D_{o\uparrow}^+ < \max_{k \neq o} D_{ok} + \epsilon.$$

Similar, if $\{\mathbf{I}_o, \mathbf{I}_\downarrow\}$ is a negative pair, the condition is:

$$D_{o\downarrow}^- > \min_{k=o} D_{ok} - \epsilon,$$

where k is a class label index and ϵ is a given margin.

These conditions support the mining of hard pairs, i.e., a positive pair where the sample still has a high distance in embedding space and a negative pair that still has a low distance in embedding space. This forces the sampling to concentrate on sampling the hard pairs. A set of positive images to an anchor image \mathbf{I}_o are denoted with \mathcal{P}_i , analog, a set of negative images to \mathbf{I}_o are denoted with \mathcal{N}_i .

Given the mined positive- and negative pairs, allows us integration into the *Multi-Similarity* loss, as derivated by Wang et al. [Wan+19b]:

$$\mathcal{L}_{\text{MS}} = \frac{1}{K} \sum_{i=1}^K \left\{ \frac{1}{\alpha} \log \left[1 + \sum_{k \in \mathcal{P}_i} e^{-\alpha(D_{ik}-\lambda)} \right] + \frac{1}{\beta} \log \left[1 + \sum_{k \in \mathcal{N}_i} e^{\beta(D_{ik}-\lambda)} \right] \right\},$$

where α , β and λ are fixed hyperparameters.

In contrast to *SL-DML*, we do not apply weighting to the classifier- and embedder loss, as no marginal improvement has been achieved in our experiments on *SL-DML*. After the model optimization, associating an action class to a query sample and set of reference samples is now reduced to a nearest-neighbor search in the embedding space. The classifier and encoder are jointly optimized.

5.5 Experiments

To show the multi-modal one-shot recognition performance, we applied our *SL-DML* approach to three datasets containing three different modalities. For our *Skeleton-DML* approach, we focus on experiments with the one-shot protocol from the NTU RGB+D 120 dataset. Results are discussed after the dataset presentation for *SL-DML* in Section 5.5.3 and for *Skeleton-DML* in Section 5.5.4.

5.5.1 Implementation

Our implementation is based on PyTorch [MBL20b], [Pas+19]. We tried to avoid many of the metric learning flaws as pointed out by Musgrave et al. [MBL20a] by using their training setup and hyperparameters where applicable. Key differences are that we use a ResNet18 [He+16] architecture and avoid the proposed four-fold cross validation for hyperparameter search in favor of better comparability to the proposed one-shot protocol on the *NTU RGB+D 120* dataset [Liu+20a]. Note, we did not perform any optimization of the hyperparameters. A batch size of 32 was used on a single Nvidia GeForce RTX 2080 TI with 11 GB GDDR-6 memory. If not mentioned otherwise, we trained for 100 epochs with initialized weights of a pre-trained ResNet18 [He+16]. The classification and metric loss were weighted by 0.5 unless stated otherwise. For the multi similarity miner we used an epsilon of 0.05 while we used a margin of 0.1 for the triplet margin loss. A RMSProp optimizer with a learning rate of 10^{-6} was applied in all experiments. If not mentioned otherwise, the embedding model outputs a 128-dimensional embedding, and for the *SL-DML* experiments the classifier yields a 128 dimensional feature vector.

5.5.2 Datasets

We used skeleton sequences from the *NTU RGB+D 120* [Liu+20a] dataset for large-scale one-shot action recognition. With 100 auxiliary classes and 20 evaluation classes, it is the largest dataset that we applied to our approach. To show the multi-modal capabilities of our *SL-DML* approach, we also used the *UTD-MHAD* [CJK15] dataset (inertial and skeleton data) and the *Simitate* [Mem+19a] dataset (motion capturing data).

The datasets are split into an auxiliary set, representing action classes used for training, and an evaluation set. In our experiments, the evaluation set does contain novel actions or actions from a novel sensor modality. One sample of each test class serves as reference demonstration. This protocol is based on the protocol proposed by [Liu+20a] for the *NTU RGB+D 120* dataset. We conducted similar experiments with the remaining two data sets. In depth descriptions are given below. First we trained a model on the auxiliary set. The resulting model estimates embeddings for the reference actions and then for the evaluation actions. We then calculate the nearest neighbor from the evaluation embeddings to the reference embeddings. This yields to which action from the reference set the current evaluation sample comes closest. Experiments for our *Skeleton-DML* approach is evaluated on the *NTU RGB+D 120* one-shot action recognition protocol.

NTU RGB+D 120 The *NTU RGB+D 120* [Liu+20a] dataset is a large-scale action recognition dataset containing RGB+D image streams and skeleton estimates. The dataset consists of 114,480 sequences containing 120 action classes from 106 subjects

in 155 different views. We follow the one-shot protocol as described by the dataset authors. The dataset is split into two parts: an auxiliary set and an evaluation set. The action classes of the two parts are distinct. 100 classes are used for training, 20 classes are used for testing. The unseen classes and reference samples are documented in the accompanied dataset repository¹. *A1, A7, A13, A19, A25, A31, A37, A43, A49, A55, A61, A67, A73, A79, A85, A91, A97, A103, A109, A115* are previously unseen. As reference, the demonstration for filenames starting with *S001C003P008R001** are used for actions with IDs below 60 and *S018C003P008R001** for actions with IDs above 60.

UTD-MHAD The UTD-MHAD [CJK15] contains 27 actions of 8 individuals performing 4 repetitions each. RGB-D camera, skeleton estimates and inertial measurements are included. The RGB-D camera is placed frontal to the demonstrating person. The IMU is either attached at the wrist or the leg during the movements. No one-shot protocol is defined. Therefore, we defined custom splits. We started with 23 auxiliary classes and evaluated with reduced training sets. We evaluated our approach by moving auxiliary instances over to the evaluation set. By this, we decreased the training set while increasing the evaluation set. In a third experiment, we evaluated the inter-joint one-shot learning abilities of our approach. For actions with IDs up to 21 the inertial unit was placed on the subject’s wrist and for the remaining IDs from 22-27 the sensor was placed on the subject’s leg. This allows us to inspect the one-shot action recognition transfer to other sensor positions by learning on wrist sequences and recognize on leg sequences with one reference example. We always used the first trial of the first subject as reference sample and the remainder for testing. Finally we evaluated the inter-modal capabilities of our approach. The model is trained on a set of training samples from a modality, i.e., skeleton sequences and evaluated on a set of samples from an other sensor, i.e., the inertial measurements. A single reference sample for each sensor is provided for our inter-modal experiments.

Simitate Furthermore, we evaluate on the Simitate dataset. The Simitate benchmark focuses on robotic imitation learning tasks. Hand and object data are provided from a motion capturing system in 1932 sequences containing 26 classes of 4 different complexities. The individuals execute tasks of different kinds of activities, from drawing motions with their hand-over to object interactions and more complex activities like ironing. We consider one action class of each complexity level as unknown. Namely the following actions, *zickzack* from *basic motions*, *mix* from *motions*, *close* from *complex* and *bring* from *sequential*. Resulting in an auxiliary set of 22 classes and 4 evaluation classes. The corresponding first sequence by filename is used as reference sample.

¹<https://github.com/shahroudy/NTURGB-D>

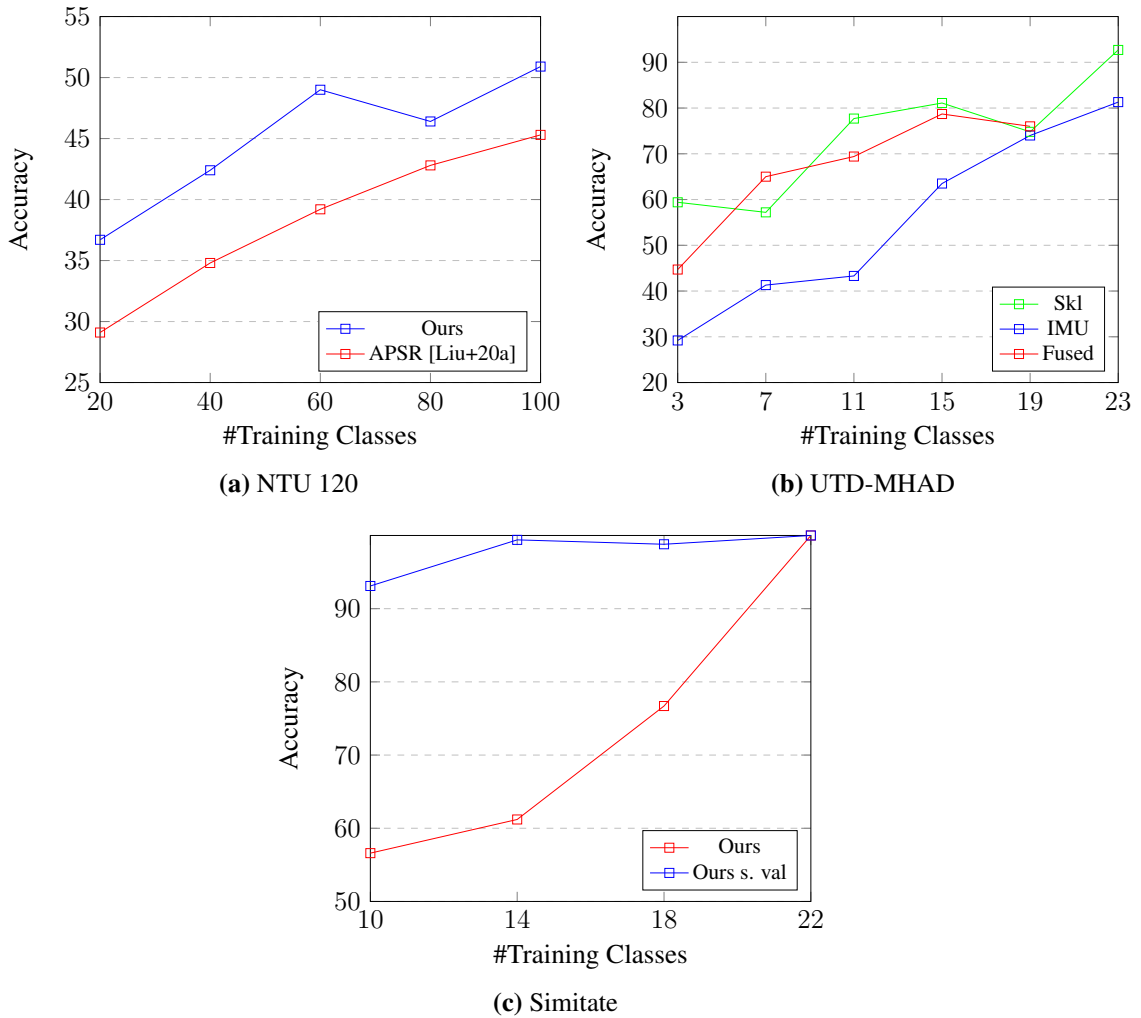


Figure 5.8: Result graphs for the NTU RGB+D 120 dataset (a), the UTD-MHAD dataset (b) and the Simitate dataset (c). *Skl* denotes skeleton data, *IMU* denotes inertial data, *Fused* denotes multi-modal data consisting of inertial- and skeleton data *s. val* denotes a static evaluation set.

5.5.3 Signal-Level Deep Metric Learning Experiments

In the following, we present the experiments conducted on the presented datasets for the *SL-DML* approach. Like Liu et al. [Liu+20a], we also experimented with the effect of the auxiliary set reduction. Results for this experiment are given in Fig. 5.8a and Table 5.1. Further we inspect the influence of different loss weighting parameters and compare two miners: Triplet Margin [SKP15] and the Multi Similarity Miner [Wan+19b] in Table 5.2.

Table 5.1: Results for different auxiliary training set sizes for one-shot recognition on the NTU RGB+D 120 dataset.

#Train Classes	APSR [Liu+20a] [%]	SL-DML ($\alpha, \beta = 0.5$) [%]
20	29.1	36.7
40	34.8	42.4
60	39.2	49.0
80	42.8	46.4
100	45.3	50.9

Table 5.2: Ablation study for our proposed one-shot action recognition approach on the NTU RGB+D 120 dataset.

Miner	α	β	Accuracy [%]
Triplet Margin [SKP15]	1.0	0.0	50.6
Triplet Margin [SKP15]	0.0	1.0	40.4
Triplet Margin [SKP15]	0.5	0.5	50.5
Multi Similarity [Wan+19b]	1.0	0.0	52.2
Multi Similarity [Wan+19b]	0.0	1.0	40.4
Multi Similarity [Wan+19b]	0.5	0.5	50.9

NTU RGB+D On the NTU RGB+D 120 dataset, we compare against the proposed baseline APSR by Liu et al. [Liu+20a]. Table 5.1 shows the results with an auxiliary set size of 100 action classes and a evaluation set size of previously unseen 20 action classes. Our proposed approach performs 5.6% better than the first follow-up [Liu+20a] and 8% better than the second follow up [Liu+18b]. Figure and Table 5.1 show results for an increasing amount of auxiliary classes (100 auxiliary classes and 20 evaluation classes are considered as the standard protocol). Overall, our approach performs better as the baseline on all conducted auxiliary set experiments. In this context, the high accuracy with 60 auxiliary classes and that the 20 additional classes added in the 80 classes auxiliary set added confusion, has to be highlighted. With 60 classes, our approach performs 9.8% better than the baseline approach with a same amount of auxiliary classes. Further, our approach performs better, with just 60% of the training data, than the first

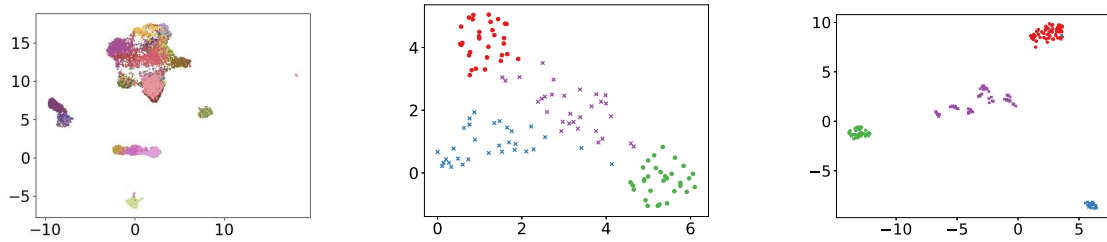


Figure 5.9: UMAP embedding visualization for the one-shot experiments using the NTU RGB+D 120 (a) dataset, the UTD-MHAD dataset (IMU) (b) and the Simitate dataset (c).

Table 5.3: One-shot action recognition results on the UTD-MHAD dataset.

#Train Cl.	#Val Cl.	SkI. [%]	Inertial [%]	Fused [%]
23	4	92.7	81.3	90.2
19	8	74.8	74.0	76.0
15	12	81.1	63.5	78.7
11	16	77.7	43.3	69.4
7	20	57.2	41.3	65.0
3	24	59.4	29.2	44.7

follow-up with the full amount of auxiliary classes. With only 40% of the training data, our approach performs comparably good as the second and third follow up. These experiments strongly highlight the quality of the learned metric. In Fig. 5.14, we show UMAP [McI+18] visualizations that give an insight about the discriminative capabilities. Distances in embedding space capture the number of identities well. This is the case for the three top clusters containing the actions (*grab other person’s stuff*, *take a photo of other person* and *hugging other person*). The two clusters at x-axis around -7.5 correspond to the actions *arm circles* and *throw*, suggesting that actions with clear high joint-relevance can also be clustered well. The most bottom cluster corresponds to the class *falling* and supports this hypothesis. On the left we have a quite sparse cluster reflecting highly noisy skeleton sequences from multiple classes. Mainly sequences with multiple persons, especially with close activities like hugging, resulted in noisy data. In that case the skeleton-estimation approach seems not be able to estimate multiple skeletons. Table 5.2 gives an ablation study showing the influence of the loss weighting and the underlying triplet mining approach. A multisimilarity miner [Wan+19b] with

Table 5.4: Inter-joint one-shot action recognition results on the UTD-MHAD dataset.

#Train Classes	#Val Classes	Train Joint	Accuracy [%]
21	6	Left wrist	80.4
6	21	Left leg	28.3
6	6	Left wrist	18.8
6	6	Left leg	59.7

Table 5.5: Inter-modal one-shot action recognition results on the UTD-MHAD dataset.

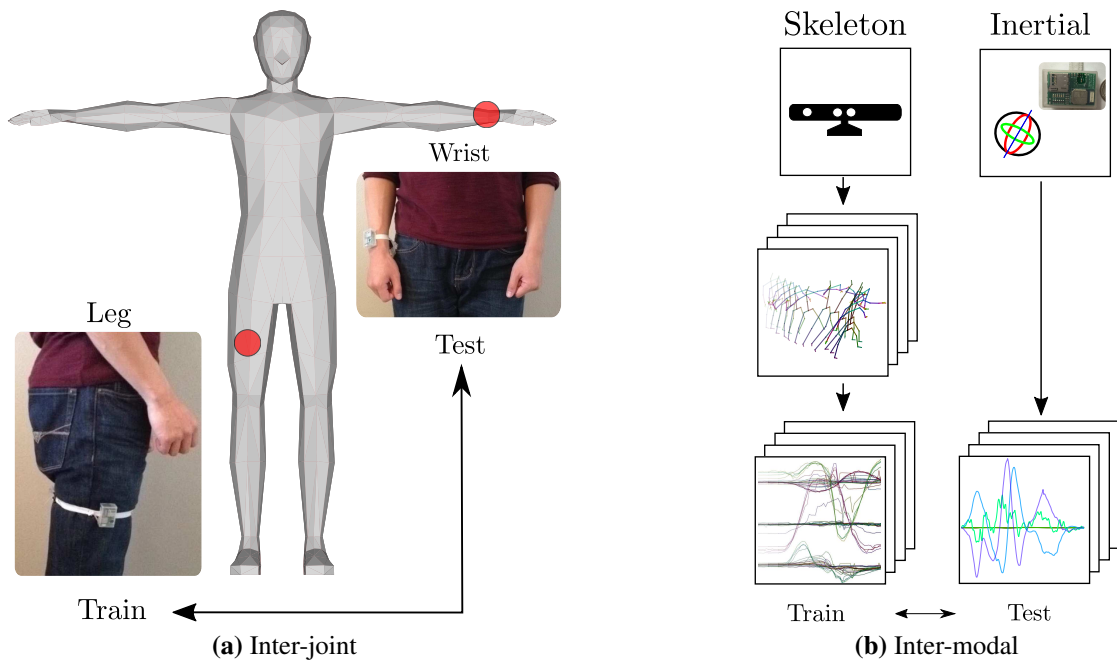
Train Modality	Val. Modality	Representation	Accuracy [%]
Skeleton	Inertial	Sparse	35.5
Inertial	Skeleton	Sparse	40.5
Skeleton	Inertial	Dense	23.1
Inertial	Skeleton	Dense	40.5

a metric loss yields the best results for one-shot action recognition on the NTU 120 dataset. In our ablation study, the loss weighting with $\alpha = 1.0$ and $\beta = 0.0$ yielded the best results for both miners.

UTD-MHAD The UTD-MHAD dataset was used to show the generalization capabilities of the proposed approach across different modalities. Results for the inter-joint experiment on inertial data are provided in Table 5.4. For the fused experiments, we concatenated $\mathbf{S}_{\text{fused}} = (\mathbf{S}_{\text{imu}} | \mathbf{S}_{\text{skl}})$, where \mathbf{S}_{imu} denotes the inertial signal matrix and \mathbf{S}_{skl} denotes the skeleton signal matrix. Concatenation is only possible with equal column matrices. Therefore, we subsampled the modality with the higher signal sample rate. By considering a signal level action representation, we could compare skeleton and inertial results and also perform multi-modal, inter-joint and inter-modal experiments. Figure 5.8b shows the effect on the resulting one-shot accuracy with increasing auxiliary set sizes. In this experiment series, we could observe that a higher number of classes used for training, not necessarily leads to a higher accuracy. This was the case for our experiments on inertial data, where training on only three classes shows a more

Table 5.6: One-shot action recognition results on the Simitate dataset.

#Train Classes	#Val Classes	Accuracy [%]
22	4	100.0
18	4	98.8
14	4	99.4
10	4	93.1
18	8	76.7
14	12	61.2
10	16	56.6

**Figure 5.10:** Inter-joint (a) and inter-modal (b) experiment setup.

similar action embedding. This observation could not be transferred to the skeleton experiments on this dataset. The selection of auxiliary classes used for training should be well-chosen. Adding more classes does not necessarily mean higher similarity in the embedding but can also add more confusion. Our inter-joint experiments yielded

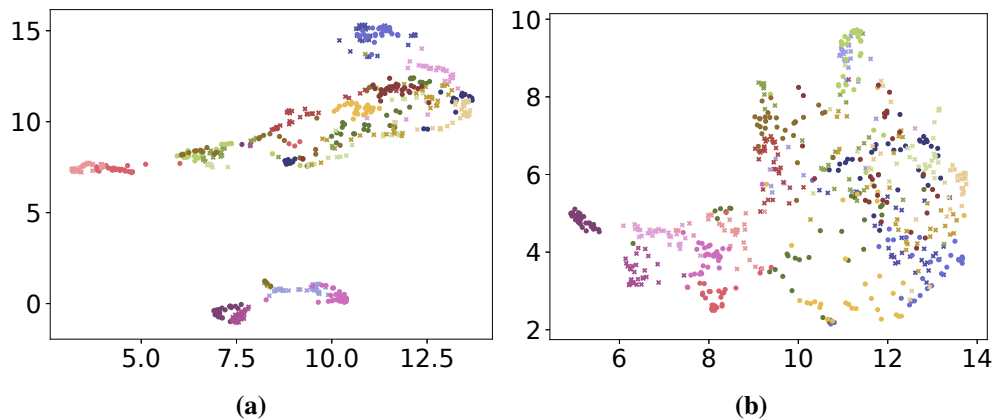


Figure 5.11: UMAP embedding visualization for inter-modal experiments. Trained on skeleton inertial measurements and validated on skeleton (a). Trained on skeleton sequences and validated on inertial measurements (b).

more transferable embeddings by training on data from the wrist and validating on the leg, as shown in Table 5.4. This holds true for our conducted experiments, but we do not want to exclude the possibility of finding a subdivision of the wrist auxiliary set that results in a higher transferable embedding. A key-insight among our experiments is that balanced classes for training and testing yielded mostly higher accuracy for lower dimensional modalities like IMU (see Table 5.3) and motion capturing (see Table 5.6). This is especially visible in our inter-joint experiments (see a depiction in Fig. 5.10a), as shown in Table 5.4. In comparison, our experiments applied to skeleton sequences benefited from more auxiliary classes (see Table 5.1, 5.3 and Fig. 5.8a & 5.8c). The conducted fusion experiments, by the concatenation of skeleton sequences and inertial measurements, show good performance in some experiments. The lower performing modality can also negatively impact the performance. This observation suggests adding sensor confidences into the approach as a future research direction. Sensor data fusion on a signal level by a single stream architecture remains an interesting and functional alternative to multi-stream architectures.

UTD-MHAD Inter-Modal In our inter-modal experiments (see Fig. 5.10b), we used all actions from one modality as auxiliary set and evaluated the other modality with a single reference sample. Results for this experiment are given in Table 5.5 and Fig. 5.11 visualizes the corresponding UMAP embeddings. The resulting one-shot recognition for inertial to skeleton performs by a large margin better (+17.4%) better than the reverse direction using our novel representation. In approximately 40.5% of the time, an action trained on a different data modality on the UTD-MHAD dataset could be recognized with just one reference sample. The signal representation from Section 3.3.3,

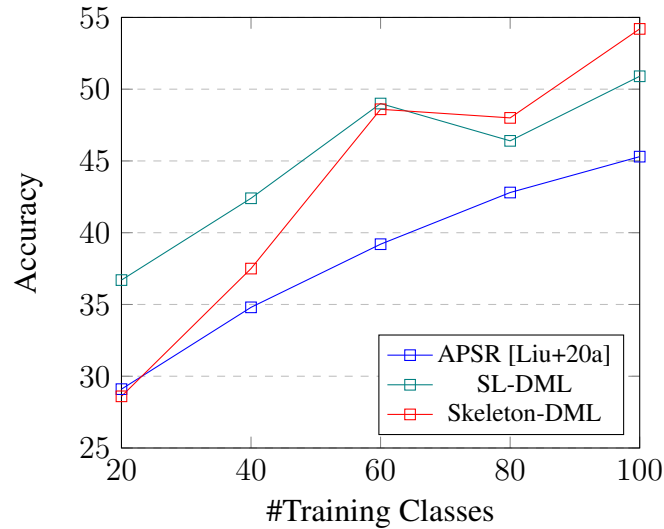


Figure 5.12: Result graph for increasing auxiliary set sizes.

generalizes better in this aspect. Overall, the inter-modal experiments show the flexibility of our proposed approach, but are subject to further improvement. We observed that the inertial measurements have a relation to the arm and hand movements of the skeleton, which explains the good transferability across the modalities.

Simitate Finally, we evaluated our approach on the Simitate dataset with motion capturing data. Results are given in Table 5.6 and Fig. 5.8c. The number of classes is comparable to the number from the UTD-MHAD dataset. The effects of the auxiliary set reduction are en-par with the experiments conducted on the UTD-MHAD dataset. Therefore, the proposed approach transfers also good to motion capturing data. The class-distances from the motion capturing experiments are higher in embedding space than the ones gathered by the inertial experiments (see Fig. 5.8b & 5.8c).

5.5.4 Skeleton-DML Experiments

We used skeleton sequences from the *NTU RGB+D 120* [Liu+20a] dataset for large-scale one-shot action recognition with the representation proposed in Section 5.4.2. Examples for sequences and their representations are given in Fig. 5.13.

The dataset is split into an auxiliary set, representing action classes that are used for training, and an evaluation set. In the one-shot protocol, the evaluation set does only contain novel actions. One sample of each test class serves as reference demonstration. This protocol is based on the one proposed by [Liu+20a] for the *NTU RGB+D 120* dataset. First we trained a model on the auxiliary set. The resulting model transforms

Table 5.7: One-shot action recognition results on the *NTU RGB+D 120* dataset.

Approach	Accuracy [%]
Attention Network [Liu+17b]	41.0
Fully Connected [Liu+17b]	42.1
Average Pooling [Liu+18b]	42.9
APSR [Liu+20a]	45.3
TCN [Sab+21]	46.5
<i>SL-DML</i> (ours)	50.9
<i>Skeleton-DML</i> (ours)	54.2

Table 5.8: Results for different auxiliary training set sizes for one-shot recognition on the *NTU RGB+D 120* dataset in %.

#Train Classes	APSR [Liu+20a]	<i>SL-DML</i>	<i>Skeleton-DML</i>
20	29.1	36.7	28.6
40	34.8	42.4	37.5
60	39.2	49.0	48.6
80	42.8	46.4	48.0
100	45.3	50.9	54.2

skeleton sequences encoded as an image representation into embeddings for the reference actions and then for the evaluation actions. We then calculate the nearest neighbor from the evaluation embeddings to the reference embeddings. As the embeddings encode action similarity, we can estimate to which reference samples the given test sample comes closest. Besides the standard one-shot action protocol and experiments with dataset reduction, we give an ablation study that gives a hint on which combination of embedding size, loss, transformation, and representation are yielding best results with our approach. Further, we integrated various related skeleton-based image representations that have been previously proposed for action recognition into our one-shot action recognition approach to compare them.

One-shot action recognition results are given in Table 5.7. Like Liu et al. [Liu+20a] we also experimented with the effect of the auxiliary set reduction. Results are given in Fig. 5.12 and Table 5.8. In addition, we analyze different representations in Table 5.10

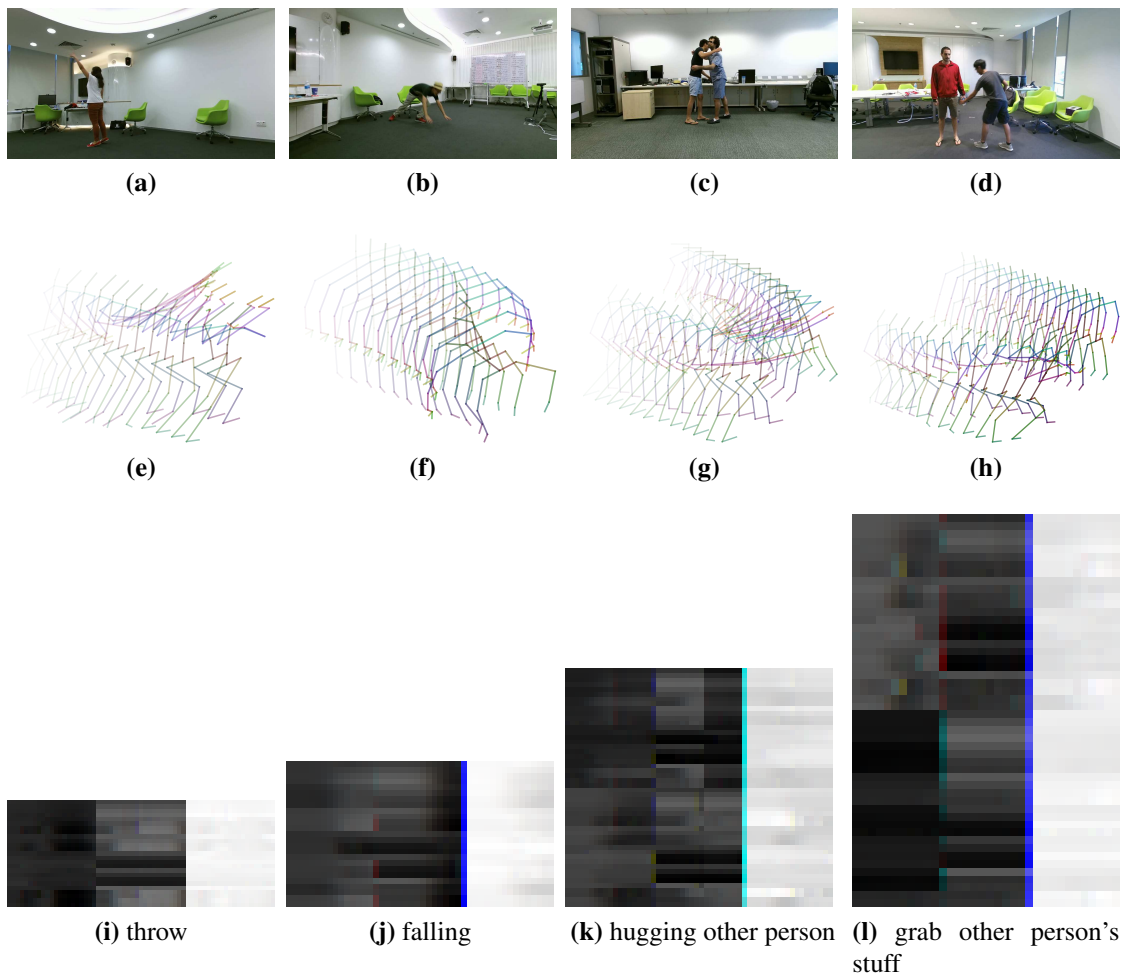


Figure 5.13: From top to bottom: A RGB Frame, the corresponding skeleton sequences and the image representation of those sequences are shown. The latter is used in our one-shot action recognition approach. The first two sequences contain single person activities, whereas the remaining two contain two person interactions. The *grab other person's stuff* sequence was shorter than the *hugging other person* sequence.

and the influence of different embedding vector sizes, metric losses and augmentations on two representations more detailed in Table 5.9.

Training Set Size Reduction

An interesting question that comes up when evaluating one-shot action recognition approaches is how much training classes are required to get a certain performance. Liu et al. [Liu+20a] already proposed to evaluate the one-shot action recognition approach

Table 5.9: Ablation study for our proposed one-shot action recognition with different representations, embedding sizes, losses and augmentations. Results are given for a training over 200 epochs. Units are in %.

Representation	128	256	512	Transform	Loss
SL-DML	55.2	50.6	52.7	None	MS
SL-DML	51.5	51.7	54.0	None	TM
SL-DML	51.8	55.3	55.8	Rot	MS
SL-DML	53.6	54.8	55.5	Rot	TM
Skeleton-DML	54.7	51.5	53.1	None	MS
Skeleton-DML	47.5	51.9	54.0	None	TM
Skeleton-DML	55.3	58.0	58.6	Rot	MS
Skeleton-DML	56.0	55.1	56.1	Rot	TM

with varying training set sizes. Aligned with Liu et al. [Liu+20a] we use training sets containing 20, 40, 60, 80 training classes while remaining a constant evaluation set size of 20. For practical systems, where only a limited amount of training data is available, this evaluation can give an important insight about which performance can be achieved with lower amounts of provided training data. It is also interesting to observe how an approach performs when adding more training data. Table 5.8 and Fig. 5.12 give results for different training set sizes for *SL-DML* [MTP20b], *APSR* [Liu+20a] and our *Skeleton-DML* approach, while remaining a static validation set. With just 20 training classes, our approach performs comparably to the *APSR* approach. With a small amount of training classes, the *SL-DML* approach performs best. In our experiments, *Skeleton-DML* performs better when providing a larger training set size. At a training set size of 60 classes, our approach performs comparably well to *SL-DML*. With 80 classes in the training set, our approach starts outperforming *SL-DML*. It is interesting to note that, aligned with the results from *SL-DML*, our approach seems to be confused by the 20 extra classes that are added to the 60 classes.

Ablation Study

To distill the effects of the components, we report their individual contributions. We examine influence of the representation, augmentation method and different resulting embedding vector sizes. Inspired by Roth et al. [Rot+20] we experiment with different embedding vector sizes of 128, 256, 512. In addition, we included the *SL-DML* representation, compare a Triplet Margin loss (TM) and a Multi-Similarity loss (MS) and

Table 5.10: Ablation study for different representations.

Representation	Accuracy [%]
Skepxel [LAM19]	29.6
SkeleMotion Orientation [Cae+19]	34.4
SkeleMotion MagnitudeOrientation [Cae+19]	39.2
TSSI [Yan+19]	41.0
Gimme Signals [MTP20a]	41.5
SkeleMotion Magnitude [Cae+19]	44.4
SL-DML	50.9
Skeleton-DML	54.2

included an augmentation with random rotations of 5° . In total, 24 models were trained for this ablation study. We trained these models for 200 epochs, as we expected longer convergence due to the additional augmented data. Results are given in Table 5.9. In the table, we highlight important results. We highlight interesting results by different colors in the table (the best result without augmentation (55.2%), embedding size of 128 (56.0%), embedding size of 256 (58.0%), TM loss (56.1%), overall, MS loss, augmentation, embedding size of 512 (58.6%)). For *SL-DML* the augmentation had a positive influence with higher embedding vector sizes of 512. Whereas the augmentation with embedding sizes of 128 only improved with the TM loss. With the MS loss and a low embedding size, the augmentation did lower the result. For our *Skeleton-DML* representation, the augmentation improved the results throughout the experiments for both losses. The best results without augmentation were achieved by the *SL-DML* representation with an embedding vector of size 128 and a MS loss. The overall best results were achieved with a MS loss and embedding vector size of 512 and augmentation by rotation using the *Skeleton-DML* representation, which improved the results of +4.4% over our approach under a comparable training setup as *SL-DML*.

Comparison with Related Representations

To support the effectiveness of our proposed representation in a metric learning setting, we compare against other skeleton-based image representations. We use the publicly available implementation for the *SkeleMotion* [Cae+19], *SL-DML* [MTP20b], Gimme Signals [MTP20a] and re-implementations of the *TSSI* [Yan+19] and *Skepxels* [LAM19] representations to integrate them into our metric learning approach. These representa-

tions have been described in Section 6.2 more detailed.

The overall training procedure was identical, as all models were trained with the parameters as in Section 5.5.3. The experiment only differed in the underlying representation. Results for the representation comparison are given in Table 5.10. While most of the representations initially target action recognition and are not optimized for one-shot action recognition, they are still good candidates for integration in our metric learning approach. We did not re-implement the individual architecture proposed by the different representations but decided to use the ResNet18 architecture for better comparability.

Our *Skeleton-DML* approach shows the best performance, followed by *SL-DML*. The *SkeleMotion* Magnitude [Cae+19] representation transfers well from an action recognition setting to a one-shot action recognition setting. Interesting to note is that the *SkeleMotion* Orientation [Cae+19] representation, while achieving comparable results in the standard action recognition protocol, performs 10% worse than the same representation encoding the magnitude of the skeleton joints. An early fusion of Magnitude and Orientation on a representation level did not improve the Skelemotion representation but yields a result in between both representations. Similar observations have been made in [MTP20b] by the fusion of inertial and skeleton sequences. The lower performing modality adds uncertainty to the resulting model in our one-shot setting.

A *UMAP* embedding of all evaluation samples is shown in Fig. 5.14 for our *Skeleton-DML* approach. Our approach shows better capabilities in distinguishing the actions *throw* and *arm circles*. In our approach, these clusters can be separated quite well whereas *SL-DML* struggles to discriminate the two classes.

Result Discussion

We evaluated our approach in an extensive experiment setup. Aside from lower performance on lower amounts of classes for training, our approach outperformed other approaches. For fair comparison, we report the result of +3.3% over *SL-DML* for training with 100 epochs and without augmentation, as under these conditions the *SL-DML* result was reported. With augmentation and training for 200 epochs, we could improve the baseline for +7.7%. Our approach leans an embedding model that captures semantic relevance from joint movements well. For example, *Skeleton-DML* differentiates successfully between activities that primarily contain hand- or leg-movements. Interactions between multiple person and single person activities are also separated well. Activities with similar joint movements contribute to are still challenging. These are the activities that are formed by the main cluster in Fig. 5.14.

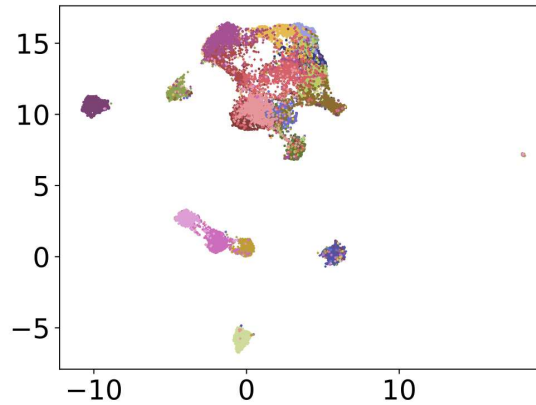


Figure 5.14: UMAP embedding visualization for our approach. Classes are: drink water ●, throw ●, tear up paper ●, take off glasses ●, reach into pocket ●, pointing to something with finger ●, wipe face ●, falling ●, feeling warm ●, hugging other person ●, put on headphone ●, hush (quite) ●, staple book ●, sniff (smell) ●, apply cream on face ●, open a box ●, arm circles ●, yawn ●, grab other person’s stuff ●, take a photo of other person ●.

5.6 Conclusion

In this chapter, we presented two approaches, *SL-DML* and *Skeleton-DML* for one-shot action recognition using deep metric learning. We propose to transform signal or skeleton sequences into an image representation. Similar to popular approaches for face recognition [SKP15], we then transform the representations. On the image representations, an embedder is learned which projects the images into an embedding vector. Distances between encoded actions reflect semantic similarities. Actions can then be classified, given a single reference sample, by finding the nearest neighbor in embedding space.

SL-DML is evaluated on three different, publicly available, datasets. Most importantly, we showed an improvement of the current *state-of-the-art* for one-shot action recognition on the large-scale NTU RGB+D 120 dataset by only using 40% of the training data. To show the transfer capabilities, we also verified our results using the UTD-MHAD dataset for skeleton and inertial data and the Simitate dataset for motion capturing data. Inter-joint experiments show inertial sensor attached to the wrist and the leg from the UTD-MHAD dataset. Motivated by the flexible underlying problem formulation, we proposed a new, challenging, inter-modal evaluation protocol that matches current trends towards multi-task more generalizable architectures. The inter-modal experiments allow training action recognition approaches on one modality and can be transferred given a single sample from a different modality. The results are objective for further enhancements that target the learning of more general models. During the non-intermodal experiments, we found that more classes used during training for lower

variate sensor data like IMUs and motion capturing systems do not necessarily improve the one-shot recognition accuracy. A good selection of training classes and a balanced training and validation set improved results across all modalities. Our *SL-DML* approach allows one-shot recognition on all the modalities we experimented with and indicates to serve as a flexible framework for inter-joint and even inter-modal experiments. By training on one modality and inferring on an unknown modality, the novel inter-model protocol can potentially shape future evaluation protocols.

For the second part of this chapter, we focussed on skeleton-based one-shot action recognition. The presented *Skeleton-DML* approach mainly differs in the underlying representations. In an extensive experiment setup, we compared different representations, losses, embedding vector sizes and augmentations. Our representation remains flexible, and yields improved results over *SL-DML*. Additional augmentation by random 5-degree rotations have shown to improve the results further. We found the overall approach of transforming skeleton sequences into image representations for one-shot action recognition by metric learning a promising idea that allows future research into various directions like finding additional representations, augmentation methods or mining and loss approaches. Especially in robot applications, one-shot action recognition approaches have the potential to improve human-robot-interaction by allowing robots to adapt to unknown situations. The required computational cost for our approach is low, as only a single image representation of the skeleton-sequence needs to be embedded by a comparably slim ResNet18-based embedder.

Chapter 6

Action Segmentation

This chapter presents an approach for the segmentation of actions in skeleton sequences. Similar to the approaches presented in Chapter 3 and Chapter 5, we represent the motion of the joints in images, but in contrast, this chapter focuses on the action segmentation problem.

Action segmentation tackles the problem of estimating the start- and end-times and the action-class labels of sequential data, in our case skeleton sequences. Action segmentation is closely related to action recognition. A naive approach it can be reduced to a frame-wise classification problem. Transformer networks have recently shown to perform well in the processing of sequential data. With DETR [Car+20], they have been applied successfully to object detection tasks. Therefore, we propose to use transformer networks for the segmentation of actions. Our approach is evaluated on the PKU-MMD dataset, with an extensive comparison of various image representations. The proposed approach can segment actions in skeleton sequences with a high class accuracy of over 95% on the test set, while the GIoU only reaches around 75%. Qualitative results show that timing estimates are often still reasonable.

This chapter is based on our publication [HMP21], co-authored with with Simon Häring of his master thesis under my supervision. Simon Häring significantly contributed to the technical realization, experiments, and preparation for the manuscript while I significantly contributed to the general idea, research concept and further contributed to the experiment preparation and literature review.

6.1 Introduction

Cameras and storage become more accessible and are distributed in masses over customer devices, leading to increasing the importance for video analysis. Many practical applications for video analysis like in elder-care, surveillance or autonomous driving

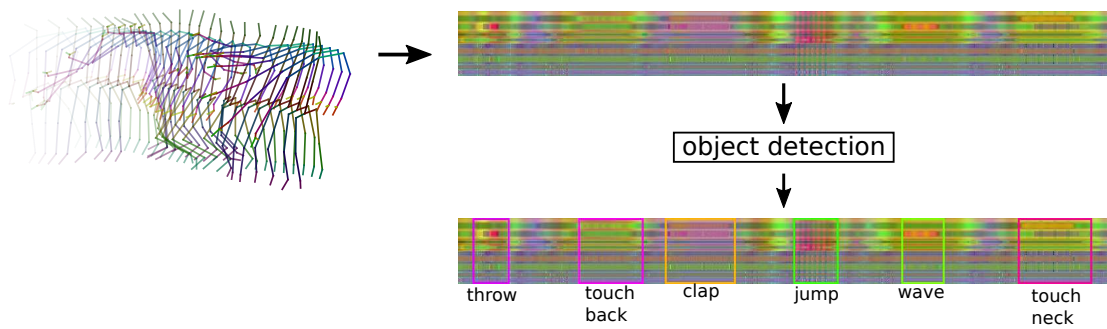


Figure 6.1: Skeleton sequences are represented as images. A transformer network is trained to segment actions from the image representation.

generate continuous image streams required to be automatically annotated to allow for automatic decisions or generate more interpretable reports.

Similar to the two previous chapters, in which we represent motions in images to associate a single class label per sequence, we utilize the representation idea but in a segmentation context of untrimmed sequences.

Surveillance and health care could use systems that scan live video for suspicious actions or people who need help. Scenarios like these include people falling or indicating sudden pain from stomach cramps or heart attacks by touching respective body parts. Further, human-robot-interaction can be improved by allowing robots to recognize actions continuously and act accordingly, i.e., to offer assistance. Particularly, action segmentation allows additional analysis of the action, like measuring its duration or the time between connected actions that belong to some longer action sequence.

We use an off-the-shelf network for object detection, namely the detection transformer DETR [Car+20] and re-purpose it for human action segmentation. DETR addresses object detection with a simple model lacking hand-made components. Instead, it combines a CNN with a transformer network [Vas+17]. These attention-based transformers are designed to process long sequences and large sets, modeling arbitrary long-distance relations. Other than Recurrent Neural Networks, which maintain and update a memory vector used to reference information from earlier in the sequence, transformers act on the entire sequence at once and generate connections between different elements when they are necessary.

In order to use this existing network for human action segmentation, we need to convert this task into object detection on images. We present and compare different ways of representing skeleton sequences as images. Unlike [Cae+19] and [LLC17], we constrain ourselves to three channel or gray-scale images to keep compatibility with DETR and numerous other architectures. We use skeleton sequences instead of videos because they are more compact and simplify processing long action sequences. The extracted skeletons are also invariant to changing background or lighting in the original

video, which means, however, that any contextual information is lost as well.

Our representations are constructed from different building blocks. These include normalization techniques, different coordinate encodings and image assembly methods like those in [Yan+19] or [MTP20a]. Our goal is to find the best-performing combinations of the described techniques and provide an insight into which properties of a representation are beneficial for training. Figure 6.1 presents an example of our method. Different actions and movements appear as different texture patterns in the image representation. This is especially noticeable for the *jump* action in the middle of the image representation. The position and shape of each bounding box encodes the time-interval of the action. This is possible because time is encoded spatially in the image.

6.2 Related Work

In the context of this thesis, we concentrate on temporal action segmentation methods that infer which action takes when place in sequential data like RGB- or skeleton sequences. Referring to in this thesis, action segmentation is not to confuse with video segmentation methods that aim at separating objects of interest in videos.

Action segmentation in videos Dense trajectories of feature vectors have been used to capture local motion information for the foreground and background motion [Wan+13]. Improved Dense Trajectories (IDT) [WS13] aim at estimating camera motion by separating background and human motion. Human motion often leads to inconsistent feature matching. Those mismatches are avoided by potential filtering mismatches using a human detector, leading to more descriptive trajectories. Incorporating IDT in a sliding window approach by Shu et al. [SYS14] enables a naive approach for temporal action segmentation. Analyzing IDT by statistical length and language models in a dynamic programming optimization has shown as an improved method to represent the temporal and contextual structure [RG16]. Kuehne et al. [KGS16] presented an end-to-end approach for action segmentation and recognition. Fisher Vectors are used to represent local feature descriptors. Those vectors are reduced by a Principal Component Analysis (PCA) to allow modeling with a Hidden Markov Model (HMM).

Lea et al. [Lea+17] presented Temporal Convolutional Networks (TCNs) by a hierarchy of temporal convolutions to perform action segmentation. The integration of TCNs is demonstrated in an encoder-decoder and by dilated convolutions. TCNs are applicable for the fine-grained time series segmentation of various sensor modalities.

Wang et al. [Wan+16] presented Temporal Segment Networks (TSNs), an approach to learn a video representation that captures long-term temporal information. Further, a hierarchical aggregation scheme allows for action segmentation in untrimmed videos. Lin et al. [Lin+19] proposed a Boundary-Matching network that offers a solution to further estimate confidence scores of densely distributed temporal action proposals.

Action segmentation in skeleton sequences Li et al. [Li+16] presented a Joint Classification-Regression RNN for skeleton sequences avoiding a typical sliding window approach by using a LSTM subnetwork that captures complex long-range temporal dynamics. First, a classification network is pre-trained for frame-wise action recognition. Then, a regression model is incorporated to capture temporal features for action segmentation. This regression network can also be used to predict future actions before their occurrence.

Datasets We now present an excerpt of interesting datasets for the temporal action segmentation task. The Breakfast dataset [KAS14] is a popular dataset for segmenting actions in videos. The dataset consists of 52 participants executing ten common cooking activities in 18 different kitchens. Similar, the recently released EPIC-Kitchen [Dam+21] contains annotation for action recognition and action segmentation from an egocentric perspective with a total of 100 hours of video material from various kitchens. The MMAct dataset by Kong et al. [Kon+19] contains untrimmed sequences and annotated action segments with RGB videos, accelerometer, gyroscope, and orientation data. This chapter uses the PKU-MMD dataset [Liu+17a] which contains annotated action segments for videos and skeleton sequences in three different views. The dataset is described in detail in the experiments Section 6.4.1.

6.3 Approach

Transformer networks [Vas+17] have recently become popular for their break-throughs in natural language processing solving long term relations in sequences by an attention mechanism. Even so transformers are commonly used for sequence classification the actual set-based formulation allows transfer to other domains as well. We convert action segmentation on skeleton sequences into object detection on image representations.

6.3.1 Detection Transformer - DETR

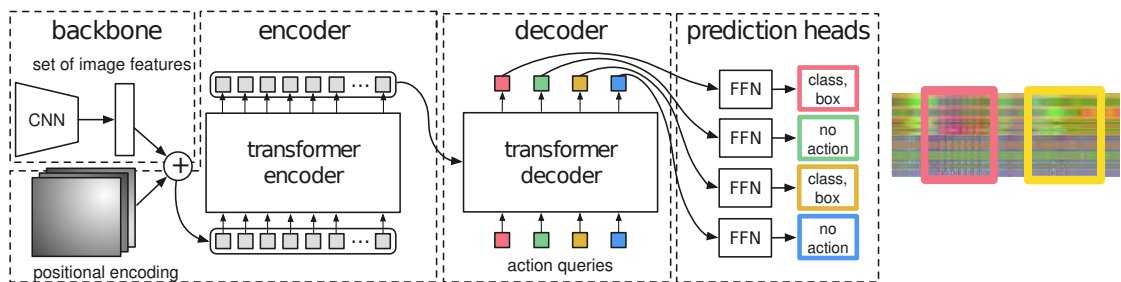


Figure 6.2: Action segmentation approach using DETR. The figure is adapted from [Car+20].

Object detection can be understood as a set prediction problem. An object detector generates a set of bounding boxes with class probabilities when given an image as its input. Just like sequences, sets can be predicted using transformers by evaluating multiple queries — one for each element in the output set. Carion et al. [Car+20] present an end-to-end object detection pipeline based on the transformer architecture by Vaswani et al. [Vas+17]. An adapted approach overview of DETR is given in Fig. 6.2. DETR is built on top of a standard CNN classifier as backbone for image data like ResNet50 or ResNet101 [He+16]. The 2048 feature maps generated by the CNN are reduced to 512 and flattened to a sequence of 512 dimensional vectors. A positional encoding vector of the same length is added to each input vector before the entire sequence is processed by the transformer. The sequence length is proportional to the image size. The encoder and decoder blocks of the transformer in DETR are the same as those presented by Vaswani et al. [Vas+17]. The input to the first decoder layer is a set of object queries which are unique, learned vectors. Each of these corresponds to one object prediction in the final output. The output vectors of this final decoder stage are independently transformed by a small feed-forward network into bounding box coordinates and class probabilities, including a *no-object* class. The additional class is required as the number of object queries is constant and usually much larger than the expected number of objects in any single image. These object queries are a set of learned vectors. Unlike anchors or region proposals, they do not explicitly encode a spatial region.

6.3.2 Representation

Skeleton sequences as they are generated by the Kinect v2 RGB-D camera contain up to six skeletons per frame. Each skeleton is defined by the 3D coordinates of its 25 joints. We construct our image representation by combining different techniques from the categories' normalization, feature extraction and image assembly.

Normalization

To achieve location-invariance, we normalize a skeleton sequence by shifting each skeleton such that its hip's center is the origin. We refer to this as *normPos* or *nP*. Alternatively, we shift each skeleton such that the mean position of the hip joint is in the origin (*nPM*). Using this, we keep the movement information and only correct for fixed offsets. Scaling is done in a later step to constrain each feature to $[0, 255]$ for dense representations (see section 6.3.2). Similar to [LAM19] and [LLC17] we also use some joints to define a coordinate system relative to the mean skeleton of a sequence. The mean-hip defines the origin, and the x -axis is given by a vector pointing from joint 17 to joint 13 in the pelvis. The z -axis is defined as the up-vector of the camera, and the y -axis is $\mathbf{y} = \mathbf{z} \times \mathbf{x}$. All vectors are normalized. We will refer to this rotation-invariant normalization as *nRM*.

Feature Extraction Extracting additional information from skeleton sequences and explicitly providing it to neural networks can make them easier to train [Cae+19]. Other than the position p of the skeleton joints, we extract their velocity v , the angles a between their adjacent edges and the rates of change of those angles A . The angles are only generated for joints with two or more adjacent edges in a depth-first tree traversal order adapted from [Yan+19]. This leads to a total of 42 angles per skeleton. In the PKU-MMD dataset, two skeleton sequences are available for each video. If *both* are used, the representation images generated for each are stacked vertically.

Image Assembly We mainly focus on a dense image representation in which each column contains one frame such that time progresses from left to right. Each row holds one measurement of one joint, i.e., the y -velocity of joint five or the x -position of joint 23 etc. We order the joints either by their ID or by the tree structure described by Yang et al. [Yan+19]. The latter is referred to as *TSSI* in the following sections. Other than these dense representations, we use sparse to convert a skeleton sequence into an image like described in Section 3.3.3. For the sparse representation, joints and their axes are considered as signals and are graphed into an image. These representations are denoted with *sparse*.

Coordinate Encoding This section further details the way that the joint coordinates or velocities can be encoded in the dense representations shown above. An intuitive way of expressing three coordinates in a three-channel image is to encode each one in a separate color channel. We denote this as *RGB*, despite using OpenCV, which defaults to a BGR color order, to generate our representations. Figure 6.3 shows an example of a nP *RGB* pv skeleton representation of the action *clapping*. The upper half is a stack of the 3D positions of all 25 joints, while the lower half contains their velocities. The rows of the first joint are black, as its position and velocity is set to zero during the normalization (it is moved to the origin and stays there). When one-dimensional features like joint angles are added to this representation, each one is repeated to fill all three channels. Therefore, the resulting image contains a mixture of colored and gray-scale regions. Instead of stacking the (x, y, z) coordinates along the channel dimension, one could stack them along the joint dimension, forming a gray-scale image. We chose two orderings, called *grayJ* $[x_1, y_1, z_1, \dots, x_n, y_n, z_n]$ and *grayC* $[x_1, \dots, x_n, y_1, \dots, y_n, z_1, \dots, z_n]$. The names stand for *gray* image with grouped *joints* and grouped *coordinates* respectively. While *RGB*, *grayC* and *grayJ* are mutually exclusive, either of them may be combined with *TSSI* to influence the skeleton joint order. In either case, including *RGB*, the values are mapped onto the range from $[0, 255]$ to fit into an 8-bit image. This is done separately for each type of feature, as the range of values for angles differ from that of positions or velocities.

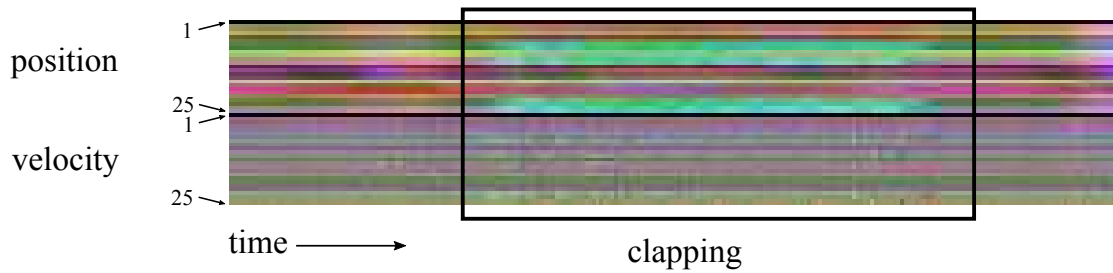


Figure 6.3: A section of a nP RGB pv skeleton representation. The variation corresponds to a single clap, where the joint IDs related to the hand and arm contribute most to.

6.4 Experiments

In our experiments, we show that an off-the-shelf object detector can be used for human action segmentation. We use the PyTorch [Pas+19] implementation of DETR, OpenCV for generating the representation images and the official PKU-MMD evaluation script for our results.

6.4.1 PKU Multi-Modality Dataset - PKU-MMD

The PKU-MMD dataset is a large-scale multi-modal dataset for human action recognition [Liu+17a]. It contains 1076 untrimmed RGB, depth and infrared videos as well as skeleton sequences, captured by three Kinect v2 cameras from different viewpoints. The untrimmed sequences are annotated with a set of action labels containing start and end frames, as well as one of 51 class labels. From the 51 action classes, 41 actions are related to daily activities and 10 classes concentrate on interactions between humans. This makes the dataset ideal for action segmentation. The authors propose to use evaluate the overlapping ratio between the predicted interval and the ground-truth interval with a threshold ω . The Mean Average Precision of Actions (mAP_{action}) gives the average precision over all action categories. Similar, the Mean Average Precision of Videos (mAP_{video}) is averaged over all videos.

6.4.2 Implementation

Unless otherwise noted, we only use the first of possibly two skeletons available in the video for our experiments. We retrain the DETR weights provided by Carion et al. [Car+20] with each representation on an Nvidia GTX 1080 for 2000 epochs with a learning rate of $3e-5$, a class error weight of 8, a bounding box weight of 20, a DICE weight of 1, a relative weight of the no-object class of 0.1, a GIoU weight of 20 and no learning rate reduction. Furthermore, we also use the cardinality loss, with weight 0.07, which compares the number of predicted objects to the actual number of objects in the

Table 6.1: Results for dense representations using different coordinate encoding methods.

Encoding	mAP _{action}	mAP _{video}
<i>RGB</i>	12.4	13.0
<i>grayC</i>	45.1	46.1
<i>grayJ</i>	28.7	29.5
3×RGB	44.6	45.0

Table 6.2: Results with the standard Kinect v2 and the TSSI joint orders and varying normalization methods.

Normalization	3×RGB		<i>grayC</i>	
	mAP _{action}	mAP _{video}	mAP _{action}	mAP _{video}
Kinect v2				
raw values	44.6	45.0	45.1	46.1
<i>nP</i>	45.4	46.0	42.1	41.7
<i>nPM</i>	39.0	40.4	47.6	48.7
<i>nRM</i>	40.4	40.6	49.7	51.8
TSSI				
<i>nP</i>	53.2	54.6	44.6	46.4
<i>nRM</i>	56.3	58.0	51.3	52.4

image. For each image, we let DETR predict 50 bounding boxes using its object queries, limiting the amount of estimated action segments per sequence. These hyperparameters are kept constant unless further noted. All following mean average precision scores are percentages on the cross-view split of PKU-MMD grouping action instances either per video (mAP_{video}) or per action class (mAP_{action}). Like in the DETR implementation, we use a ResNet-50 backbone which is trained alongside the transformer with a learning rate of 10e-5.

6.4.3 Results

We experimented with each of the previously presented components of our approach. Further, we compare the dense representations against *sparse* representations and *state-of-the-art* methods.

In Table 6.1 we present results for different dense representations of the skeleton position as described in section 6.3.2. The *grayC* encoding achieved best results. Comparably well is the 3 times stacked *RGB* representation, suggesting that the height width ratio of the final encoded image influences the results. We then compare different normalization strategies for the two better performing encodings *grayC* and $3\times\text{RGB}$. Table 6.2 shows the mean average precision scores for different normalization techniques using the standard Kinect v2 joint order (top) and the semantically sorted *TSSI* joint order (bottom). The *nP* had a positive effect on the Kinect v2 joint order, whereas the mean normalization for position and rotation resulted in a lower mAP. In contrast, for the *TSSI* joint order, the *nRM* normalization yields the highest scores.

Table 6.3: Ablation study modal features. Positional data (*p*) as before with versions using joint velocities (*v*), joint angles (*a*) and angular velocities (*A*) as well as stacks of these features.

Feature	height	<i>nRM TSSI 3×RGB</i>	
		mAP _{action}	mAP _{video}
<i>p</i>	147px	56.3	58.0
<i>v</i>	147px	22.9	21.6
<i>a</i>	126px	43.0	43.5
<i>A</i>	126px	9.7	10.1
<i>pv</i>	294px	67.2	68.1
<i>pva</i>	420px	76.2	77.0
<i>pvaA</i>	546px	70.2	71.4

Table 6.4: Results with two skeletons

Approach	mAP _{action}	mAP _{video}
$4\times pa$	71.7	72.0
$2\times pa$ both	77.9	79.0
hybrid	73.6	74.7

Table 6.3 shows results for different features combinations. Positions and angles perform much better than their time-derivatives. Combining the positional data with velocities and angles achieves the best results. However, adding angular velocities to this, leads to a reduction in mAP score. Disregarding the *pvaA* result, which may be explained by the low performance of angular velocities, we can observe the trend of taller representations resulting in better scores to continue.

Table 6.5: Comparison of dense- and sparse representations.

Approach	height	mAP _{action}	mAP _{video}
<i>sparse p</i>	420px	59.3	59.2
<i>sparse pa</i>	420px	58.7	59.4
<i>TSSI 6×RGB p</i>	294px	68.0	69.5
<i>TSSI 4×RGB pa</i>	364px	71.7	72.0

In order to compare the influence of the second skeleton for interaction classes, we test three representations built from stacks of positional and angular data. Figure 6.4 shows the per-class average precision of the three different representations discussed in this section. The two representations using both skeletons consistently score higher than $4 \times pa$ in interaction classes (marked with an asterisk). The large difference in mAP score between the hybrid and $2 \times pa$ both representations is mostly due to large improvements, including *falling* and *pointing at something*. Table 6.4 shows the results of this experiment. While the 364px tall $4 \times pa$ representation scores in between the 294px $3 \times pv$ and 420px $3 \times pva$ representations from Table 6.3, $2 \times pa$ both surpasses all previous representations by using the second skeleton where available. The hybrid representation does not achieve comparable results, despite also using both skeletons. Here, if only one skeleton is available, the image region is filled with a copy of the first skeleton instead of black pixels like with $2 \times pa$ both.

To show generalization of our action segmentation approach, we integrate the *sparse* representation as presented in Section 3.3.3. In Table 6.5 we compare two sparse representations with dense representations. Adding angular information to the sparse representation seems to have no effect on the result. In Table 6.6 we compare against related methods. Our approach is outperformed by the latest CNN-based approaches [Li+19b; Li+17] but performs better than the PKU-MMD dataset baselines [Li+16]. None of the previous presented approaches uses transformer networks for the action segmentation task. While the class accuracy lies above 95% regularly on the test set, the GIoU, which determines how well our method predicts the start- and endpoints of an action, only reaches values around 75%.

Fig. 6.5 gives exemplary segmentation results for three different sequences of the PKU-MMD dataset. On the bottom of each frame, the ground-truth and prediction are shown. Classes are denoted by different colors. Many predicted segments match with the ground-truth, with our approach estimating segments with slightly differences in the start- and end-times comparing to the ground-truth. When inspecting those segments in detail the timing estimates appear to be still reasonable.

In Fig. 6.6, we give exemplary segmentation results. These examples support that

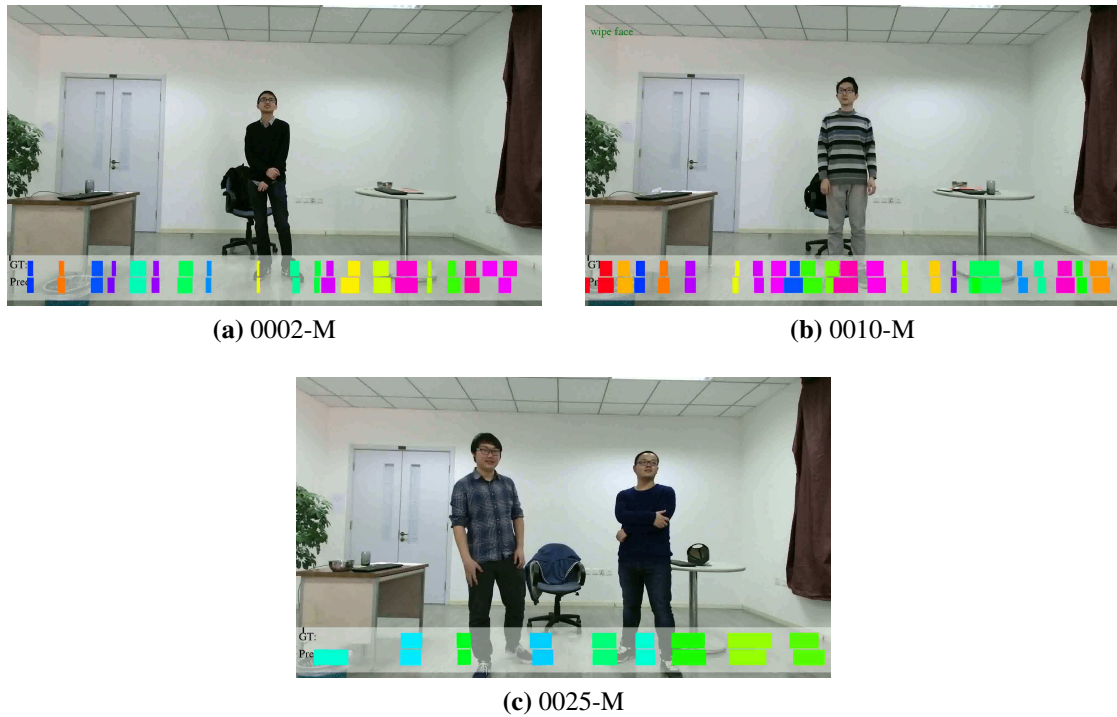


Figure 6.5: Example segmentation results for the PKU-MMD dataset.

Table 6.6: Comparison with related approaches.

Approach	mAP @ $\omega = 0.5$
Li et al. [Li+19b]	94.4
Li et al. [Li+17]	93.7
JCRRNN [Li+16]	53.3
<i>nRM TSSI 3×RGB pva</i> (ours)	77.0
<i>nRM TSSI 2×RGB pa both</i> (ours)	79.0

segmentation class estimation is correct for the majority of the segments and the lower performance falls back to varying start- and end-time predictions, which are oftentimes still reasonable.

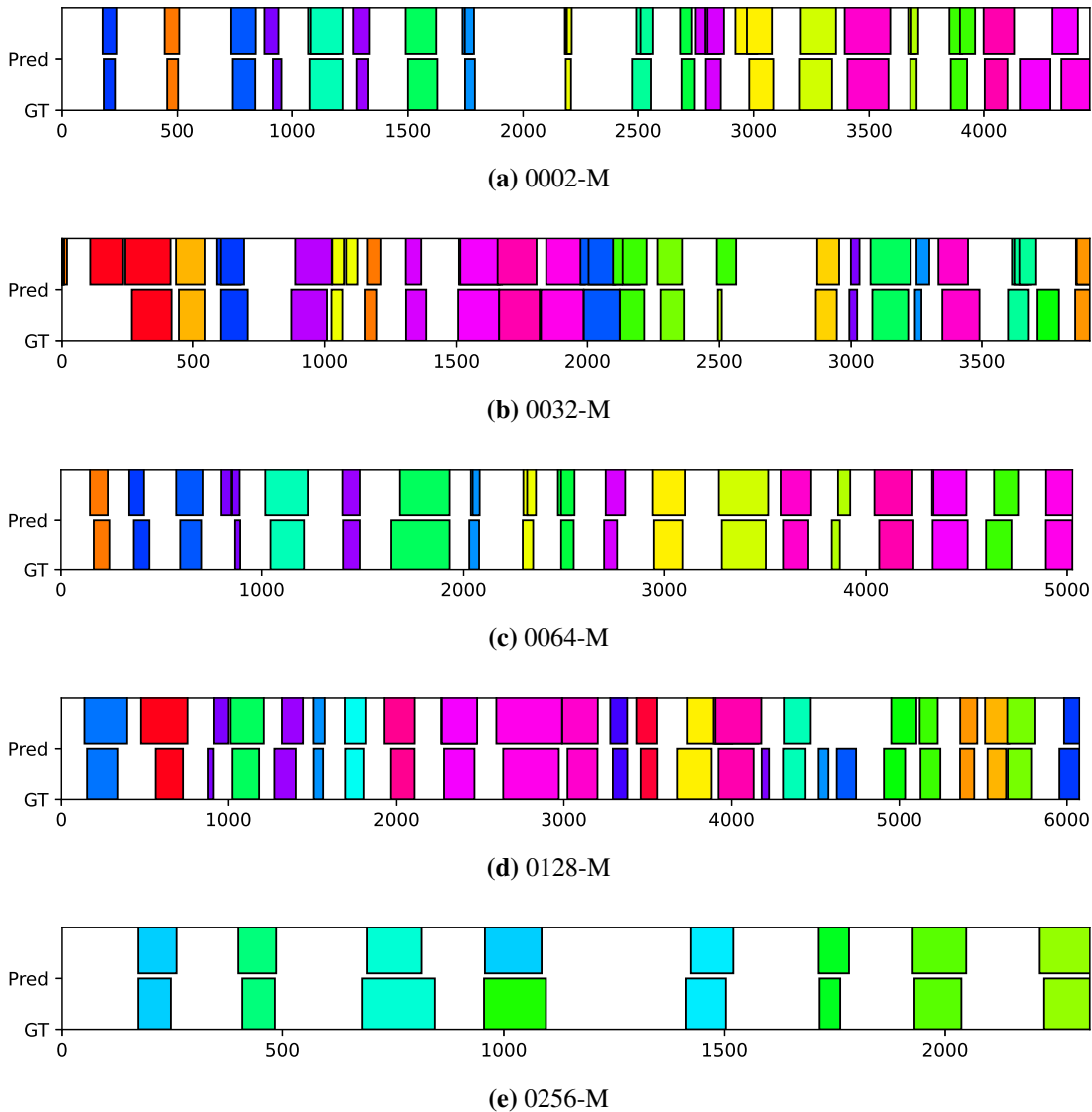


Figure 6.6: Exemplary action segmentation results on the PKU-MMD dataset. Predicted segments are at the top, Groundtruth segments are visualized at the bottom. The x-axis denotes the timing in milliseconds.

6.5 Conclusion

We presented an approach for action segmentation on skeleton sequences using a transformer-based object detector. Similar to our approaches presented in Chapter 3, we employ a variety of representations based that can be used to encode skeleton motion into an image flexibly. The action segmentation problem is then formulated similar to an object

detection problem on the image representations. We use the transformer-based object detection approach DETR to segment action sequences from the PKU-MMD dataset. Our approach reaches a high class recognition accuracy but is outperformed by *state-of-the-art* methods for skeleton-based action segmentation by lower start and end estimation of the actions.

Chapter 7

Benchmarking for Imitation Learning

While the first part of the thesis concentrated on associating class labels to sensor data sequences originating from various data sensors in a supervised and unsupervised training setting, this part aims at transferring the observed actions to a robotic agent. In detail, one aspect that we found a lack of focus in the existing literature is the benchmarking of imitation learning approaches. In its current state, most imitation learning approaches are trained and tested in complete simulated environments or on custom acquired data that prevents experiment reproduction. In this chapter, we present an approach for the benchmarking of imitation learning approaches.

This chapter extends our previous work on benchmarking for imitation learning approaches, which has been published in [Mem+19a]. We give an updated related work discussion, an extended description of the experimental setup for the dataset acquisition, and provide additional insights about the proposed metrics. Section 7.1 provides an introduction to imitation learning and benchmarking. In Section 7.2, we present literature related to imitation learning research and the benchmarking approaches from the computer vision and robotics communities. The resulting dataset is presented in Section 7.3. Details about the dataset integration, proposed metrics and the general benchmark are given in Section 7.4. Finally, we conclude this chapter in Section 7.5.

7.1 Introduction

Recent research in machine learning aims to automate an increasing amount of the system architecture to solve tasks. Learned feature maps succeeded hand-crafted- feature design [KSH12; LeC+89]. NAS [ZL17] succeeded the manual design of neural network architectures. When transferring these developments, we can argue that in the context of robotics, robots learn either from human demonstrations to succeed sequences of textual program descriptions defined by expert programmers or learn to solve a task solely on their own.

The application of robots in domestic environments is foreseeable. We argue that with the future spread of robots, the demand for custom service robot tasks and, therefore, expert programmers will increase dramatically. Thus, we publish a dataset that fosters imitation learning approaches just by visual observation of humans interacting with their environment. This supports the demonstrator when interacting naturally with its environment (let it be objects or humans). This idea stands in high contrast to current approaches that pull demonstrators out of their natural interaction by putting sensor suites or using kinesthetic teaching of robots. The Programming by Demonstration paradigm is most famous for various applications in industrial repetitive task programming.

Motivated by the increasing success of deep neural networks that recently opened up possibilities for reasonable accurate object recognition [He+16; KSH12; Sze+15], detection [Red+16; Liu+16b], semantic segmentation [He+17; BKC17; PCD15] and human pose estimations [Cao+17; Sim+17; Wei+16], we argue in advancing these fundamental approaches to actual scene understanding and even replication with a mobile domestic service robot.

We identified that one issue with imitation learning research is that approaches are often just empirically evaluated and often demonstrate performance only on qualitative results on a small action set. Reproducibility and direct comparison are not given as the observed data is not publicly available. This defines a gap between other research topics like the benchmarking of mapping approaches in robotics [GLU12; Stu+12] or benchmarks from computer vision research like image classification [KSH12], object detection [Lin+14], object tracking [BS08].

This issue might be caused by the complexity of the evaluation process for the imitation learning tasks. In general, benchmarks for robotic systems are more complex. A robot system usually is equipped with various sensors and different subsystems that fuse information from various sensors. Those sensors need to be calibrated and a ground-truth needs to be defined that many times isn't as trivial as for the image classification task. These aspects are tackled on various levels. Interesting approaches follow the *sim2real* method, where robot agents are trained in a simulated environment, and the learned policies are then transferred to real-world robot systems [Hig+17; Höf+21; Ope+19]. Sim2Real requires precise simulators modelling general physics down to a motor controller level. Training agents in a simulation carries benefits as for instance, simulation can be executed faster than real-time and can collect decades of experience in days [Kad+20] due to parallelization. However, the trained policies often do not directly transfer to real-world robots [Hig+17].

In this chapter, we present an approach for the benchmarking problem of imitation learning tasks. The benchmark entails a dataset that incorporates real-world sensor measurements from an RGB-D camera. The dataset is strongly coupled with an integration into a simulator. Further, we propose metrics for the evaluation of the imitated robot

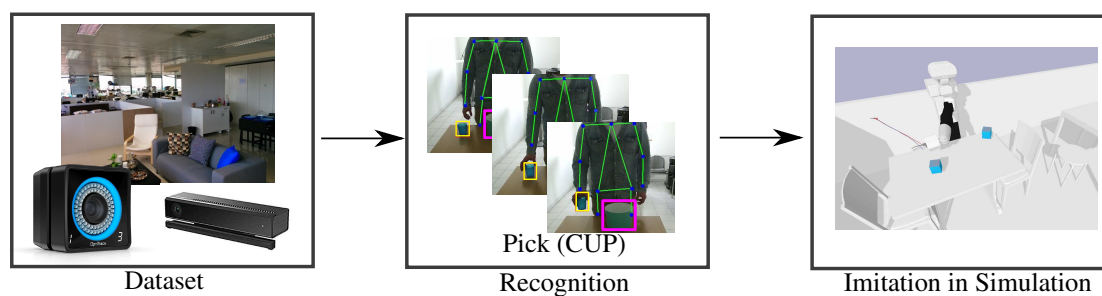


Figure 7.1: Overview: This figure gives an overview of our benchmarking model. We provide a dataset contained recorded in the real world. The sequences of these datasets are supposed to be interpreted by imitation learning approaches, which then execute the imitation in a simulated environment which is grounded by the ground truth initial object positions. After the performance in simulation, the results are evaluated.

behavior. The lack of missing metrics has already been highlighted in 2009 by Argall et al. [Arg+09] and again in 2018 by Osa et al. [Osa+18].

We found robotic imitation learning approaches that use custom collected data for experiments, but this data has not been published for general access. This makes reproduction and comparison more difficult or even impossible. In high contrast to other currently available datasets, we not only focus on the recognition of actions, but also on a more profound understanding of the interaction between humans and objects. Even though we also recorded ground truth positions of the demonstrator’s hand and the interacting objects, the goal of the benchmark is to advance in markerless visual imitation learning approaches.

Simitate will be applicable for approaches in different fields like imitation learning through reinforcement learning [Dua+17], genetic programming [GP18] or generative adversarial networks [Gai+15]. Besides imitation learning, the dataset can be used for action recognition or object tracking but does not primarily target these fields.

The main contributions of this chapter are:

- a novel publicly available dataset containing different individuals performing daily activity tasks,
- a novel benchmarking component that enables researchers to compare their results in a simulated environment,
- metrics for evaluation based on the imitated trajectory and the effect are proposed.

7.2 Related Work

Imitation Learning Most approaches use custom datasets and methods for evaluation, making direct comparisons vague. Ross et al. [RGB11] presented a supervised approach for imitation learning by dataset aggregation, called DAgger. Expert policies which gather a dataset of trajectories are used to train a second policy that aims at mimicking the trajectories well. Afterwards, more policies from expert demonstrations are used again to mimic the demonstrations, but now the trained policies are added to the dataset. The next policy is then defined as the policy that best mimics the expert on the whole dataset. Laskey et al. [Las+17] proposed an off-policy approach which injects noise into the demonstrator’s policy. By this, the demonstrator is forced to correct the injected noise and a recovery behaviour from errors can be trained. In comparison to DAgger [RGB11] they claim the approach to be faster and more robust. The data from the physical experiments on a real robot is not available. Ho et al. [HE16] presented an approach for extracting policies directly from data by a model-free imitation learning algorithm. Their approach has been proven to show the same results as inverse reinforcement learning problems. One shot imitation learning approaches [Yu+18; Pai+18; Dua+17] have recently gained popularity. Further, virtual reality approaches have been used for learning new activities by demonstration [Bat+17; Ram+14]. A promising crowdsourcing approach of human-robot interactions was proposed by Mizuchi and Inamura [MI17]. This potentially could enable learning robot activities by demonstrations through virtual reality. A direct transferability from virtual reality to real-world robots is challenging because of the usage of simulated sensor data. We try to tackle this bottleneck in this chapter. More recently, interesting Real2Sim [Sad+18] approaches train robot controllers entirely in simulation and are successfully transferred to real robots.

Datasets Comparable datasets mostly target action recognition. Therefore, the datasets presented in Section 3.4.2 are related to the imitation learning to a certain degree. However, many action recognition datasets do not focus on human-object or human-scene interactions. Weinzaepfel et al. [WMS16] presented DALY, a dataset containing ten daily activity classes found in 500 YouTube videos with a total duration of 31 hours. Pirsiavash and Ramanan [PR12] created a first person dataset containing images from people fulfilling daily activity tasks. A comprehensive survey for action recognition is given by Zhang et al. [Zha+16]. Many published datasets focusing on imitation learning target autonomous driving [Cod+18; ZC16]. Gupta and Malik [GM15] presented a dataset based on a subset of the COCO [Lin+14] dataset, targeting semantic role labeling by verbs describing people interacting with objects. An interesting dataset, named PROX, to support the analysis of humans interacting with an environment was presented by Hassan et al. [Has+19]. Similar, the GRAB [Tah+20] and ContactPose [Bra+20] datasets concentrate on humans grasping objects. Those datasets could potentially be

interesting to improve imitation learning in future, as they allow more detailed analysis of humans and their object- and environment interactions.

The dataset that comes closest to our proposed dataset is the CAD-120 by Koppula et al. [KGS13] which contains 120 different RGB-D camera sequences where four individuals perform activities like making cereals, microwaving food and more. In addition, the dataset contains skeleton data provided by a skeleton tracker and also manually annotated object tracks.

Robot Benchmarking Benchmarking nowadays enables quantitative evaluation in many research topics like autonomous driving [GLU12], object tracking [BS08; WLY13; Mil+16], and RGB-D SLAM systems [Stu+12]. Those benchmarks build a comfortable environment for evaluation as most commonly standard formats, evaluation metrics and scripts are specified for result comparison. Most of them even collect produced evaluation results online [GLU12; Mil+16] in a leaderboard. Some of the later benchmarks also integrate the replication by actual robotic systems i.e. for grasping [Lei+17]. Virtual reality environments have previously been used [Zha+18; VCS18] for evaluation of human robot interfaces.

Around the time, that we proposed Simitate, an increasing interest in robotic benchmarking was evolved. Yu et al. [Yu+19] presented a benchmark for multitask and meta reinforcement learning. They target training policies that generalize well to entirely held-out tasks. Different protocols for various difficulties are proposed. James et al. [Jam+20] presented a robot learning benchmark in a simulated environment containing 100 hand-designed tasks. Demonstrations can be generated. Both of those benchmarks concentrate on fully simulated environments without incorporation of real sensor data, neither demonstrations from human observations. Toyer et al. [Toy+20] presented an imitation learning benchmark that aims at benchmarking generalization capabilities of imitation learning approaches. Rana et al. [Ran+20] presented a benchmarking approach for Learning by Demonstration (LbD) approaches. Four different methods were benchmarked in a large-scale user experience. Demonstrations by varying expert levels were recorded. Reproductions using the four different methods from different starting positions were recorded and have been rated by mechanical turks. This benchmark relies on human observations for the evaluation of the reproductions.

Robot Challenges and Competitions In form of competitions like RoboCup@Home [Wis+09] robotic systems are benchmarked in domestic environments, however, due to the biannual changes of the rules and not fully objective opinions of referees the comparison should be seen critical. Further, the focus is set on a time constrained one shot evaluation in most tasks. In contrast, the European Robotics League [Lim+16] puts a focus on benchmarking and uses explicit metrics. However, long term benchmarking and the limited amount of participating teams still makes long-term comparability



(a) Reflective markers on hand



(b) Reflective marker on interacting object



(c) Reflective markers on RGB-D camera

Figure 7.2: Dataset setup. Reflective markers are attached on the human's hand (a) on the interacting objects (b)) and the RGB-D camera (c).

difficult. The HEART-MET challenge [TH21] follows an interesting approach to combine dataset challenges with real-world robot challenges. The World Robot Summit has several sub-challenges like assembly [Yok+19], where industrial robots are challenged in manipulation tasks. Points are given in tasks of various difficulty. As the task is well-defined by partial points for subtasks that can clearly be indicated as success or failure. The service robot category, Similar, the *Future Convenience Store Challenge* [Wad17] proposes practical tasks for service robots. Three challenges, two of them with clear metrics, like e.g., for the toilet cleaning task the cleaning rate (measured by before and after the cleaning task) and for the store and disposal task points corresponding to the stored and disposed items are proposed.

Metrics Some metrics have been proposed, mainly for the correspondence problem of imitation learning tasks [AND07]. A promising approach is to measure the effect based on [Ali+06] were demonstrated and imitated effects are compared by their displacements in relation to other objects. Most common for the evaluation of imitation learning tasks are qualitative observations [RGB11; Las+17]. This is a major deficit in comparison to other well established fields.

7.3 Dataset

In the following section, we describe the dataset recording setup for the Simitate benchmark dataset. An RGB-D-camera has been calibrated against a Motion Capturing System. The dataset sequences contain humans performing demonstrations of various complexities in a replicated domestic environment, targeting the benchmarking of domestic service robots. The resulting dataset sequences are introduced.

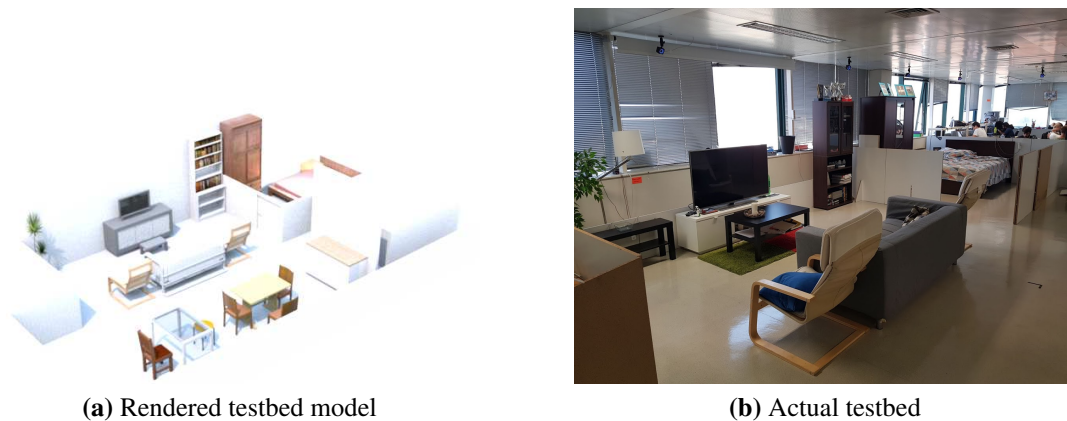


Figure 7.3: The ISRoboNet@Home testbed.

7.3.1 Setup

To record the dataset, we used a Kinect 2 RGB-D camera mounted on a tripod. Data was acquired in an exemplar apartment modelling common real-world apartments, including different furniture and rooms. 12 OptiTrack PRIME 13 cameras were mounted on the ceiling. In total, an area of $50m^2$ is covered by the system. The optical center of the RGB-D camera is calibrated against the Motion Capturing System. Rigid body markers are attached to all relevant interacting objects and the human demonstrator. The demonstrator is completely visible during recording, except when he is occluded by objects or furniture he is interacting with. The individual sequences were recorded at a number of different locations in the apartment. For inspection purposes, we also recorded a camera stream giving an overview of the apartment. The marker- and apartment-setups are shown in Fig. 7.2.

7.3.2 Testbed

The testbed ISRoboNet@Home¹ (see Fig. 7.3) has been set up for the European Robotics League to support the benchmarking of service robots. A detailed floor plan is also depicted in Appendix B. It aims at imitating a domestic environment separated in different rooms, including standard furniture and objects. The Motion Capturing System described above is integrated in the testbed and allows recording ground truth data of interacting humans, robots, as well as objects. Besides the installed Motion Capturing System this testbed has the following benefits: its initial state can be recovered, it is similar to real apartments and it is open for use by research groups. This benefits also allows everyone to extend the set of recorded sequences. In fact, we want to

¹<http://welcome.isr.tecnico.ulisboa.pt/isrobonet/>

motivate researchers to add new demonstrations or environments. The rendered model (see Fig. 7.3a) is used for the integration in the simulator to allow integration of the real-world sensor data recorded in the real environment (see Fig. 7.3b).

7.3.3 Calibration

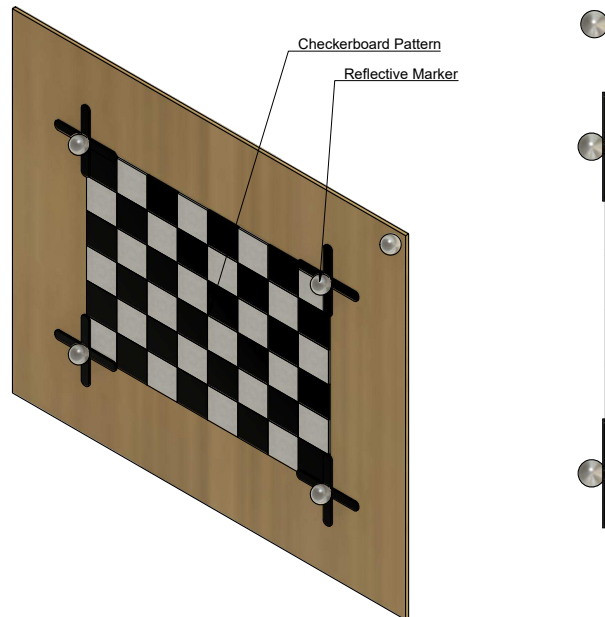


Figure 7.4: Checkerboard with reflective markers. The top right marker is used to define a unique pose. The central point on the plane in the middle of the four reflective markers on the checkerboard pattern define the centroid of the calibration pattern.

For the calibration of the RGB-D camera to the Motion Capturing System, we follow the ideas of Sturm et al. [Stu+12]. Reflective markers were attached at the corners of a checkerboard pattern. The setup for the calibration pattern and markers are depicted in Fig. 7.4. The centroid of the checkerboard was estimated using the Motion Capturing System and the central checkerboard pixel for corresponding image coordinates. It was ensured that the printed pattern was completely planar. We estimated the reflective marker height using the CW-200 marker and updated the centroid to be on the same planar surface as the printed pattern.

Our goal is to transform point clouds from the RGB-D camera into a common reference coordinate system with the Motion Capturing System such that we can project real sensor data into a virtual environment later on. We intrinsically calibrated the RGB and infrared cameras of the RGB-D camera by following a method of Zhang et al.

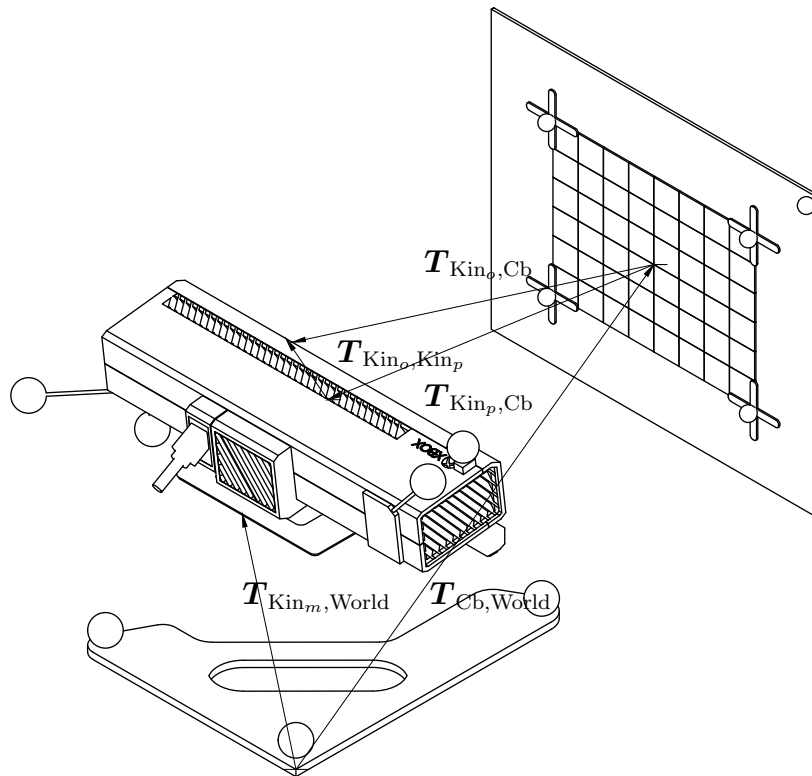


Figure 7.5: Extrinsic transformation overview. We are interested in finding the transformation between the optical center of the RGB-D camera and the RGB-D camera frame in Motion Capturing System coordinates. The Motion Capturing System calibration pattern (bottom) defines the world origin. Correspondences between the checkerboard in world coordinates and image coordinates can be generated automatically by finding the central checkerboard point in image coordinates.

[Zha00] and then extrinsically calibrate the camera’s optical center using the Levenberg-Marquardt method [Mar63]. Fig. 7.5 illustrates the given and required transformations.

Intrinsic Calibration

For the intrinsic calibration, a chessboard pattern with 7 horizontal inside corners and 5 vertical inside corners with a distance of 20 mm was used. A complete planar attachment on a wooden board was ensured. The RGB-D camera and calibration board were mounted on a tripod to ensure no motion artifacts. The RGB and infrared camera were then each presented various differing poses of the pattern. The calibration was guided by a tool reporting about the different scale and skew poses of the pattern to ensure varying poses. For each of the poses, a corner detector is used to extract features from the checkerboard.

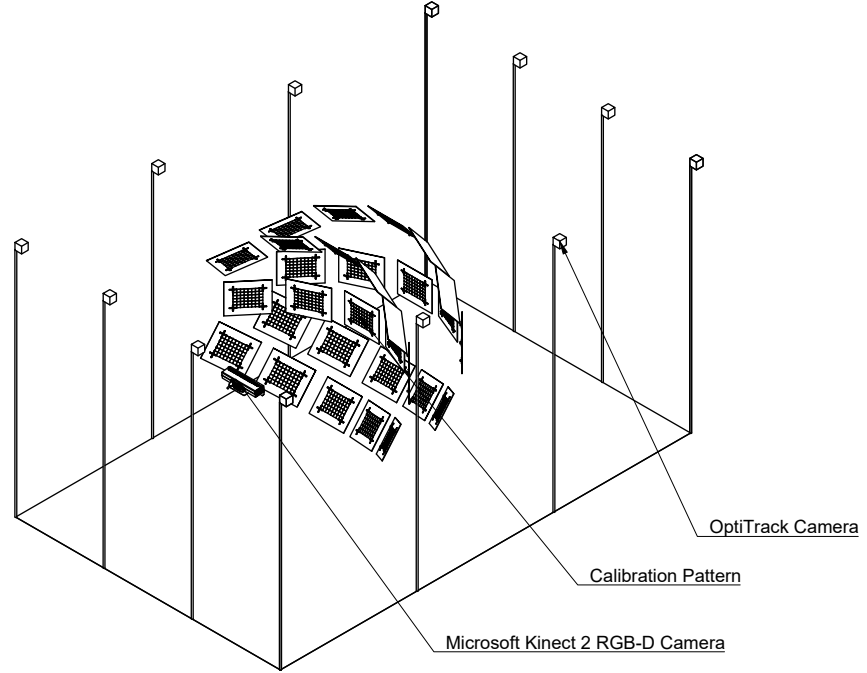


Figure 7.6: Motion Capture / RGB-D Camera Calibration setup

Our goal is to estimate the parameters of our camera matrix \mathbf{K} :

$$\mathbf{K} = \begin{bmatrix} \alpha_x & \gamma & u_0 & 0 \\ 0 & \alpha_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix},$$

where α_x, α_y denotes the focal length, u_0, v_0 denote the optical center, and γ denotes the skew. By using the method of Zhang [Zha00], the intrinsic camera parameters as well as the poses for each input pattern are given. We are interested only in the intrinsic parameters for later estimation of the extrinsic calibration between the camera and the Motion Capturing System. A closed-form solution initializes the model parameters, then the radial distortion parameters are estimated by minimizing a simplified projection equation and finally the whole projection equation is refined by a maximum likelihood estimation. For details, we refer the reader to Zhang's approach [Zha00].

Extrinsic Calibration (Camera to MoCap)

Now, given the camera matrix \mathbf{K} , we can formulate our problem as finding the translation vector \mathbf{t} and rotation matrix \mathbf{R} to transform the projective center of the RGB-D camera $\mathbf{T}_{\text{Kin}_o, \text{Cb}}$ into the marker frame of the Motion Capturing System $\mathbf{T}_{\text{Kin}_m, \text{World}}$:

Finding the transformation $\mathbf{T}_{\text{Kin}_m, \text{Kin}_p}$ is crucial for aligning the pointclouds from the Kinect with the Motion Capturing System. A method to get the rotation \mathbf{R} and translation \mathbf{t} is to solve the projective n-point problem. For an in-depth description of solutions of the projective n-point problem, we refer to Marchand et al. [MUS16]. We collect a set of synchronized correspondences of the central checkerboard point. The central checkerboard world coordinate is given by the calibrated motion capturing system. For that, we used a checkerboard setup as shown in Fig. 7.4. The overall calibration scene for collecting correspondences is depicted in Fig. 7.6. The central checkerboard point in pixel coordinates is given by the same feature detector that has been used for the intrinsic calibration already. The pixel coordinates are then transformed to homogeneous image coordinates. Note, we assume a calibrated and rectified camera here. An approach to estimate \mathbf{R} and \mathbf{t} is to minimize the re-projection error with:

$$\operatorname{argmin}_{\mathbf{R}, \mathbf{t}} = \sum_{i=1}^m \left| \mathbf{p}_{\text{Image}, i} - \mathbf{K}(\mathbf{R}\mathbf{p}_{\text{Cb}, i} + \mathbf{t}) \right|^2,$$

where $\mathbf{p}_{\text{Image}, i}$ denotes the homogeneous image coordinate and $\mathbf{p}_{\text{Cb}, i}$ denotes the central checkerboard point in homogeneous world coordinates. The minimized rotation \mathbf{R} and translation \mathbf{t} define the required transformation $\mathbf{T}_{\text{Kin}_m, \text{Kin}_p} = [\mathbf{R} | \mathbf{t}]$.

The Motion Capturing System has carefully been calibrated before recording the sequences using OptiTrack Motive motion capturing software with a CW-500 marker. A common origin has been estimated using a CW-200 marker in a fixed point of the apartment. For the motion capturing calibration, we achieved the following results. The mean overall wand error was 0.136 mm . For re-projection, we got a mean 3D error of 0.523 mm , and a mean 2D error of 0.099 pixels. The worst mean re-projection 3D error was at 0.642 mm and the worst mean 2D error was at 0.143 pixels. The RGB-D camera has been calibrated intrinsically and extrinsically.

Fig. 7.7 shows the re-projected marker of the checkerboard center. The transformation between the centroid of the RGB-D camera's rigid body and optical center of the RGB-D camera are then estimated [Zha00]. In order not to interfere with the calibration result by motion, we mounted the RGB-D camera and the checkerboard on tripods. The inverse transformation is used between the Motion Capturing System pose of the RGB-D camera and its optical center. A precise calibration is especially important for the alignment of real-world data and later imitation in simulation. Too high residuals will lead to an inaccurate alignment between simulation and real-world observation and can affect the imitation performance.

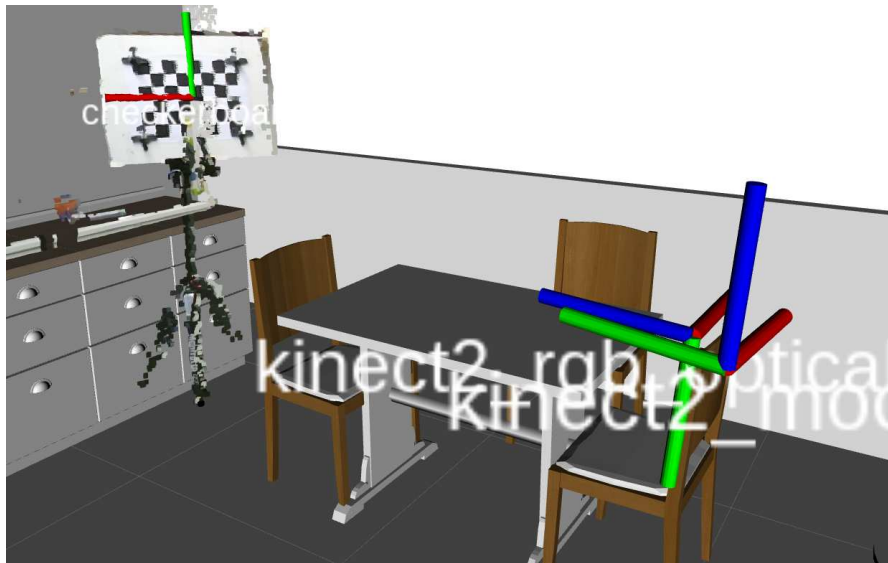


Figure 7.7: A visualization of the resulting calibration. The checkerboard center is correctly aligned to its corresponding Motion Capturing System marker.

7.3.4 Human-Object Interactions

We used common affordable home accessories that we got from a worldwide serving furniture retailer. The used objects are depicted in Fig. 7.8 and listed with their item numbers in Table 7.1. their labeled training images and pre-trained models for two widely spread recent approaches [Red+16; He+17]. The images have been labeled with support of a recent guided image segmentation approach [Man+18]. The provided data allows to easily reproduce the results and diminished the hurdles to develop approaches for this benchmark. We tried to get colorful objects too, as the focus of the presented benchmark should not be on object recognition, but on the imitation learning aspect.

We mounted rigid body markers at the back of the right hand of the demonstrator. An exemplary setup for the human is shown in Fig. 7.2a. We ensured that human pose estimates using a recent key-point detector are not interfered by the marker setup. We provide human body keypoints extracted with OpenPose [Wei+16] and projected using the depth channel into world coordinates as well.

7.3.5 Sequences

We recorded sequences for multiple purposes. First, we want to ensure that different categories of imitation learning can use this dataset. Therefore, we recorded sequences that aim at the interpretation of the demonstrations on a symbolic and on a trajectory level.



Figure 7.8: Objects used for the dataset.

Sequences on a trajectory level are further divided into cloning tasks, where the human performs a movement and the goal is to mimic the movement. More challenging sequences contain object interactions. Different individuals perform all sequences. We provide sequences that cover not only local demonstrations, but also movements between different places in the apartment of Fig. 7.3. For tasks like opening a door, we ensured to handle multiple doors of the apartment. We divide the sequences based on their level of difficulty. *Basic Motion* sequences contain drawn figures with the right hand. Its intention is to clone the observed movement. They also serve as testing sequences for the hand position estimation. *Motion* sequences contain activities like reaching for an object with the hand, picking, placing, moving or pushing it. More complex activities contain tasks that are categorized as *Complex* sequences. *Sequential* scenes contain multiple basic motions in various random combinations over a longer period of time. The complete list of sequences is given in Table 7.2. In Fig. 7.9 example, sequences are visualized. The tables in Appendix A contain one example for each class contained in the dataset.

7.4 Benchmark

We propose a combined approach of real-world observations and simulated environment for benchmarking imitation learning approaches. The initial object locations and

Table 7.1: Simitate object list

Name	Number	Type
365+	604.063.04	plates
JÄLL	202.428.90	ironing board
BITTERMANDEL	204.323.81	vase
STEKA	926.258.00	pan
PS 2002	303.879.72	watering can
LILLNAGGENA	402.435.96	Shower squeegee
FÖRDUBBLAA	903.459.41	2-piece knife set
HEAT	870.777.00	trivet
ANTAGENA	202.339.61	Dish brush
BLASKA	701.703.29	Dust pan and brush
GNARP	303.358.41	3-piece kitchen utensil set, black
SVAMPIG	602.576.05	sponge
FLUNDRA	401.769.59	dish drainer
FÄRGRIK	003.189.56	mug
VISPAD	602.575.25	colander
GLIS	800.985.83	box with lid
FEJKA	903.751.55	Artificial potted plant
GUBBRÖRA	902.257.31	Rubber spatula

positions of the observing sensors are propagated into a carefully reconstructed simulation of the testbed. This approach has multiple benefits: First, this enables evaluation methods for imitation learning and extends currently available datasets that focus on action recognition. Second, it supports generalization as the imitated behavior could be benchmarked with a wider variety of simulated robots and simplifies the transfer to real-world robots. Third, it enables generalization to verify the imitated behavior with a variety of objects and locations.

Exemplary, we provide integration into two widely used simulations [KH04; CB21] in the robotics and machine learning community. The benchmark in combination with the provided dataset therefore allows the evaluation of action recognition and task imitation on a semantic and trajectory level. As action recognition is already addressed by many other datasets, we focus on the imitation aspect in the benchmark description.

To reduce the complexity in application of this benchmarking approach and to foster the development of imitation learning approaches, we provide labeled training data

Table 7.2: Sequence overview

	# Seq	Avg. Length in s	Total Length in m
Basic Motions			
Circle	104	6.83	11.58
Rectangle	105	6.84	11.97
Heart	85	6.85	9.70
Triangle	85	6.85	9.70
Zickzack	85	6.83	9.68
Motion			
Reach	79	7.97	10.49
Move	79	7.96	10.48
Push	30	9.40	4.70
Pick	79	7.97	10.49
Place	79	7.96	10.49
Pour	224	8.25	30.83
Stack	63	14.63	15.36
Wipe	31	29.06	15.01
Mix	33	14.36	7.90
Complex			
Ironing	92	31.74	48.66
Clean	92	28.11	43.11
Throw	50	6.84	5.70
Cut	49	19.37	15.82
Open	40	9.37	6.24
Close	20	4.34	1.44
Sequential			
Rearrange	65	19.33	20.94
Pick and Place	409	14.21	96.91
Place into	60	10.67	10.67
Bring	82	21.02	28.73

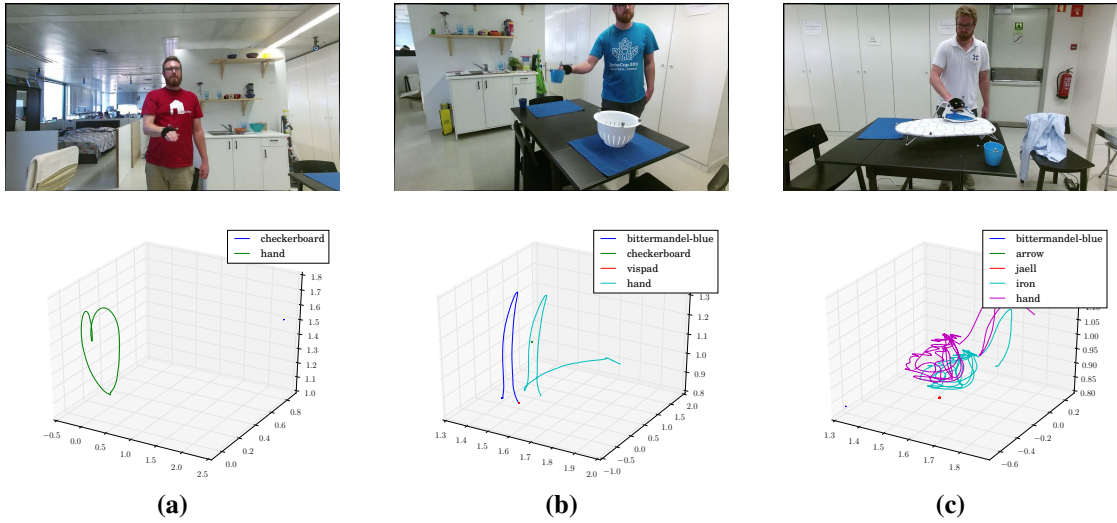


Figure 7.9: Example sequences image on top and plotted trajectories at the bottom for (a) a basic motions heart sequence, (b) a motion sequence for reaching, (c) a complex sequence for ironing.

for object segmentation and object detection as well as pre-trained models for current state-of-the-art approaches [He+17; Red+16]. The benchmark is supposed to be executed sequentially. First, the individual sequences are played back. This sequence has to be analyzed by an approach either on semantic or trajectory level. After the analysis, the task is to reproduce the observed actions. Generalization is evaluated by replication of the same tasks using different initial setups, but common actions on previously unseen sequences. In the observation step, sequences from the dataset will be analyzed and relevant information for the recognized action, interacting objects and arm trajectories should be extracted. We provide a class that simplifies this for later evaluation. The ground-truth information from the sequence is used to initially set up the virtual representation of the testbed in simulation. A simulated robot should then execute the observed action. This allows evaluation of the achieved effect and trajectory error measurements.

7.4.1 Effect

Using the effect has been proposed by Alissandrakis et al. [AND07]. We integrate effect evaluation for relative and absolute effects after performance of the imitation. Evaluating the relative object pose seems to be appropriate when objects are placed very close to each other. In this case, we can measure the Relative Pose Error (RPE) between the final object pose p_e and the relative ground truth poses between the object

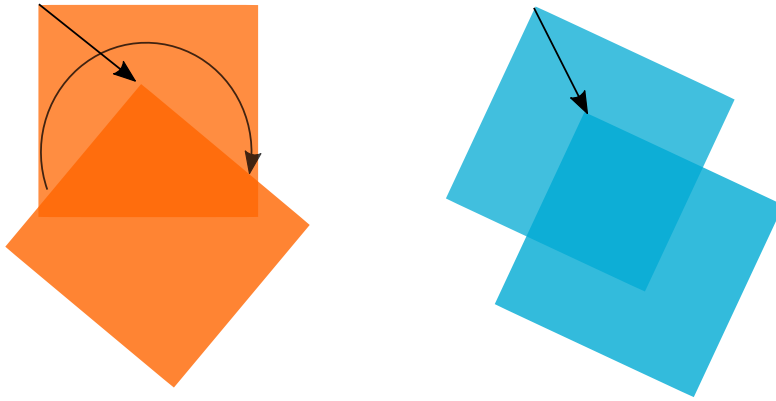


Figure 7.10: Examples for a pose error. On the left, a rotational and translational error is shown. On the right, only a translational error is shown. The rotational error is omitted for symmetric objects.

and the j -th of n surrounding objects $\mathbf{p}_{g,j}$ like:

$$\text{RPE} := \sum_{j=1}^n (\mathbf{p}_e \ominus \mathbf{p}_{g,j})^2,$$

where \ominus is the inverse motion-compensation operator [Küm+09] that can be imagined as the relative 3D transformation between two poses. This metric is inspired by suggestions for the accuracy of SLAM systems [Küm+09; Stu+12]. The success of the imitation is evaluated based on post conditions that are modeled by the end state of the ground truth. An example for the pose error is given in Fig. 7.10. In other cases, it will be more relevant to aim for an effect in the human’s coordinate frame. For that, we use the absolute pose error:

$$\text{APE} := \mathbf{p}_e \ominus \mathbf{p}_g.$$

In the proposed benchmark, we provide scripts for automatic evaluation of both metrics and weight their interest depending on the performed action. For many common everyday objects like bowls, the rotation around their z axis is irrelevant because their symmetry is not distinguishable, even for humans. In this case, we skip the angular component in the error calculation. This metric is used for *motion* sequences. Additionally, it could be applied on other sequences as well, but this is not primarily targeted by this benchmark.

7.4.2 Trajectory Error

The other metric that we propose is based on the relative trajectory error between the robot’s end-effector and the interacting object over the period ($1 : m$) of imitation. This

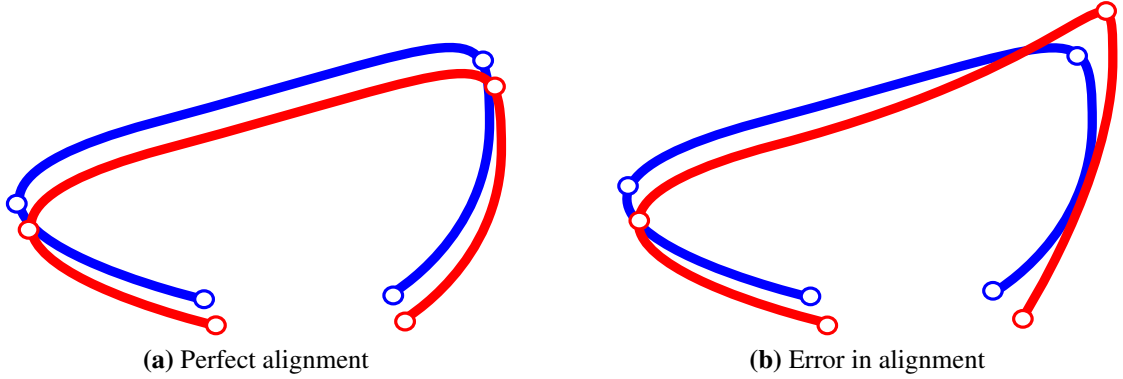


Figure 7.11: Trajectory metric visualization. Given the blue **ground truth** trajectories as observed from the demonstrator’s hand and the red **imitated trajectory** by the robot’s end-effector, in (a), the trajectory, even so that the location of the imitation is off, the trajectory could be perfectly aligned with an $\text{RMSE} = 0$ while the trajectory in (b) cannot be perfectly aligned and results in an $\text{RMSE} > 0$.

results in a similar metric as proposed in [Stu+12] for visual odometry using the Root Mean Square Error (RMSE):

$$\text{RMSE}(\text{RPE}_{1:m}) := \sqrt{\frac{1}{m} \sum_{j=1}^m \|\text{RPE}_j\|^2}.$$

By considering the relative trajectory error, the exact location and scale of the imitation is not of interest. The focus is on the quality of the imitated end-effector trajectory. This metric is considered, e.g., for the *basic motions* sequences, where no hand-object interactions take place. A visualization and analysis of exemplary trajectories is given in Fig. 7.11.

7.4.3 Baseline

To prove the validity of the proposed trajectory metrics and the benchmarking model, we implemented a simple approach for imitating human motions based on visual observation. Such a scenario is visualized in Fig. 7.12a. For showing the validity of the effect metric, we took exemplary sequences and compared them against other demonstrated sequences involving the same set of objects.

For the basic motion sequences, we evaluated the absolute trajectory error of the imitation. We use a keypoint detector for human pose estimation [Wei+16] to estimate the hand positions in every frame of the sequence. The position of the right hand is projected in 3D space by using the depth channel of the corresponding pixel. The APE

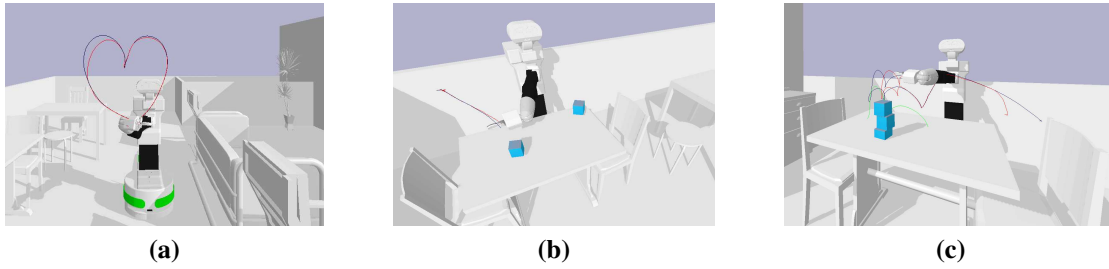


Figure 7.12: Trajectory comparison in simulation. The red line denotes the end-effector positions while the black line shows the ground truth positions of the demonstrator with a basic motions heart sequence (a), a motion pick sequence (b) and a sequential stack sequence (c).

of the first sequence of each set are with two robots, shown in Table 7.3. This table shows that the imitated hand poses with the robot’s end-effector are reasonably accurate but subject to further improvement. We show the applied metric for the approached estimated hand keypoints (KP) and also in contrast what could potentially be reachable with the proposed same initial setup by the robot with the ground-truth hand position (GT). The keypoint results are heavily influenced by outliers that occurred through projection errors of the corresponding 2D estimation to the corresponding depth value, i.e., in cases where no depth could be estimated.

We also verified the validity of the effect evaluation using the RPE for the imitation of a place sequence. Example settings are shown in Fig. 7.12b, 7.12c. The robots are placed in front of the table in a similar position as the RGB-D camera was placed. The goal is to replicate the final state of the scene. For simplicity, we attach the moved object to the end-effector position and computed the inverse kinematics to the goal location to compute the RPE. For the TIAGo robot, we got an average distance error of 0.047 m and a rotational error of 0.013 rad for the active object. The source code to reproduce the results is provided on the project page.

7.5 Conclusion

We proposed a novel benchmark for imitation learning tasks. A dataset recorded with a RGB-D camera calibrated against a motion capturing system is coupled with a simulated representation of the environment. Metrics for evaluation are proposed. The goals of this benchmark are to foster comparability, reproducibility, and the development of approaches for imitation learning tasks with a slight focus on visual imitation learning approaches. The dataset does not just contain toy examples (like reaching or moving objects) but also more complex challenges to solve, for example, ironing cloths and sequences for imitation on a trajectory level without object interactions. Simitate aims

Table 7.3: Evaluation of the absolute translation error (units are in m)

	Min	Max	Mean	RMSE
Circle				
TIAGo KP	0.006	0.673	0.105	0.160
TIAGo GT	0.007	0.108	0.034	0.038
Sawyer KP	0.011	0.755	0.110	0.174
Sawyer GT	0.003	0.333	0.030	0.041
Rectangle				
TIAGo KP	0.010	0.548	0.065	0.086
TIAGo GT	0.005	0.139	0.028	0.032
Sawyer KP	0.009	0.769	0.061	0.086
Sawyer GT	0.005	0.386	0.026	0.033
Triangle				
TIAGo KP	0.015	0.382	0.068	0.094
TIAGo GT	0.007	0.112	0.024	0.034
Sawyer KP	0.014	0.400	0.078	0.106
Sawyer GT	0.007	0.114	0.025	0.036
Heart				
TIAGo KP	0.011	0.362	0.054	0.073
TIAGo GT	0.007	0.083	0.027	0.033
Sawyer KP	0.010	0.701	0.057	0.085
Sawyer GT	0.005	0.184	0.030	0.037
ZickZack				
TIAGo KP	0.024	0.213	0.072	0.081
TIAGo GT	0.001	0.098	0.036	0.043
Sawyer KP	0.022	0.214	0.070	0.079
Sawyer GT	0.002	0.108	0.035	0.043

at keeping the entrance barrier low by providing a complete suite with datasets, pre-trained models, integration into widely spread simulations and simple visual baseline approaches as a starting point. It can be extended by adding new tasks using an openly accessible testbed. The effect metric will come to a limit on imitation learning tasks with soft-bodies like bedsheets or liquids.

Chapter 8

Conclusion and Outlook

In this thesis, we introduced approaches for the recognition of actions on signal streams. Next to supervised and semi-supervised training settings, we presented methods for the segmentation and fusion of various sensor-data-streams. In addition, we presented an imitation learning benchmark that goes beyond the sole recognition of actions, but towards the imitation of observed actions in a robotic system. We put a particular emphasis on comparability and reproducibility, therefore, all of our presented approaches are evaluated on large-scale public available datasets and the source code is made available. At the time of writing, our one-shot action recognition approaches *SL-DML* and *Skeleton-DML* hold the *state-of-the-art* on the challenging NTU RGB+D 120 one-shot action recognition protocol by a significant margin. Further, our *Fusion-GCN* approach outperforms other approaches on the large-scale multi-modal MMAAct dataset by a significant margin. Our CNN-based approach for action recognition on various sensor that aims at generalizing well across sensor data modalities is outperformed by recent GCN-based approaches, that focus on skeleton sequences, when dealing with high number of action classes.

First, we presented methods for representing motion originating from various sensor modalities in images. By formulating the action recognition problem on a signal-level, our method remains flexible to modalities ranging from skeleton sequences, IMU, Wi-Fi -CSI fingerprints over motion capturing data to RGB-sequences (after transformation into a human pose feature space). The motion representations are then trained using a recent EfficientNet-CNN model in a supervised training setting. Our results suggest good generalization capabilities across different sensor-modalities without adaptations of the underlying representation or network architecture. Being generalizable across different sensor modalities is a huge practical benefit over other available approaches that often focus on improving results for a single sensor modality.

Further, we presented an approach for multi-modal action recognition based on GCNs into a skeleton graph. Various modalities are fused into a skeleton graph on two dimensionality-levels in an early fusion scheme, either on a channel dimension or a

spatial dimension. Early fusion schemes are particularly interesting as they add limited complexity to a base model, contrarily to late fusion schemes which introduce separate streams for each additional modality.

For the fusion on a channel dimension, additional modalities are fused by introducing additional node attributes. On a spatial dimension, additional nodes are incorporated into the skeleton-graph.

We demonstrate *state-of-the-art* performance on the MMAAct dataset, where the incorporation of additional modalities has further improved the superior performance of the GCN.

For our experiments, we observed that the fusion by inertial measurements from a smartwatch improves the skeleton-graph while to many modalities decrease the action recognition performance. This suggests future research towards the modelling of uncertainty into the training process, for instance by weighting different modalities with the loss.

The first two presented approaches focused on generalization or the fusion of various sensor data modalities. However, it might also be favorable to generalize to previously unseen actions in practice. Methods that are able to recognize actions given only a few samples are beneficial for further applications, e.g., to improve human-robot-interaction, but are also of interest, e.g., for the recognition of anomalies. We proposed to transform motion representations into an embedding space that encodes action similarity. The embedding model jointly optimizes the embedding space with features of a backbone CNN and a metric learning loss that optimizes for self-, positive-, and negative-pair similarities. By continuing to follow the signal level problem formulation, our semi-supervised action recognition approach generalizes well across different sensor modalities. Our metric-learning based one-shot action recognition approaches achieve *state-of-the-art* performance on the challenging NTU 120 one-shot action recognition task, even with just a fraction of 40% of the available training data. The models are learning semantic contributions of the skeleton-joints. A highlight is a novel inter-modal action recognition protocol that is a result of our flexible problem formulation. Concepts from action samples trained on one modality can be transferred to a novel, unseen modality with just a single reference sample. Inter-modal action recognition remains highly practical due to its flexibility and aligns well with the current research efforts towards multi-task models.

In this thesis, we also addressed the segmentation of actions in skeleton-streams. We formulate the action segmentation problem as an object detection problem. We propose to use transformer networks, in detail the DETR, and their abilities to model long-term attention to address the action segmentation problem for skeleton sequences. Various representations have been evaluated with the approach. Experiments were conducted on the PKU-MMD dataset. We showed that detection transformers are good candidates for the action segmentation task. Performance is especially good for the recognition of

the actions but is outperformed by *state-of-the-art* methods for skeleton-based action segmentation by lower start and end estimation of the actions.

Finally, we presented a hybrid imitation-learning benchmark. The imitation learning problem goes beyond the recognition of actions but towards its imitation of robotic systems. Currently, imitation learning approaches are widely evaluated only in simulated environments or in toy scenarios, with datasets mostly not publicly available. With the Simitate benchmark, we aim at making the imitation learning task more accessible and comparable. For that, we recorded a dataset where different individuals perform daily tasks of various complexity levels, ranging from basic motions to manipulation activities like picking or placing, to complex tasks like ironing. The dataset provides RGB-D-sequences that are calibrated against a Motion Capturing System. This setup allows pairing real sensor data with ground-truth motion to train a policy that imitates the observation. Having access to the ground-truth data of the demonstrator's hand and the interacting objects gives many possibilities for the training and evaluation of imitation learning approaches in a simulated environment. Beyond the commonly used success rate for the evaluation of agent-based systems, we proposed metrics that give insights into the imitation in the simulated environment either on a trajectory-level, where the focus is on accurately imitated trajectories of the end-effector, or on an effect-level, where the final state of the manipulated objects and their relation is in focus.

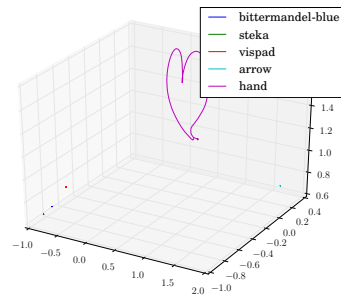
The research topics related to action recognition and imitation contain a lot of potential for future research that would exceed the course of the thesis. We now present some research ideas for future research. With minor adoptions, our action segmentation approach could be applied for the spatio-temporal action localization problem. Pose sequences from multiple persons would be represented in an image. Naively, the proposed action segmentation approach could be executed for pose sequences per person and per sample. The pose sequence enclosed by the segmentation would allow for a pose level spatio-temporal action localization. Calculating the bounding box of the pose at a given time step would yield a bounding box of the person and action class label that could be re-projected into image space from the information contained in the representation. NAS is an interesting research direction that suggests novel research topics for action recognition. First, sampling all existing architectures in an architecture space would be engaging for the evaluation of action recognition. Having access to trained models in an architecture space would allow for in-depth architecture studies. From our perspective, the inter-modal action recognition protocol is a promising candidate for future research and improvement, which requires the modeling of generalization capabilities. Reformulating the NAS optimization target to minimize the error on a validation set of a different modality should lead to architectures that generalize better across various modalities. An interesting future research direction leading requiring generalization of neural networks in its core might also be the incorporation of prediction methods for test errors into the NAS routine. Currently, most NAS approaches aim at finding ar-

chitectures that optimize the number of floating-point operations while achieving high validation accuracies. Recently proposed methods for the estimation of generalization capabilities would be great candidates to experiment with a reformulation of the optimization target, to minimize the predicted test-error while minimizing the floating-point operations. There are many datasets for action recognition of various modalities already existing. However, there is a lack of action recognition datasets that includes laser range data, which could potentially verify to which extent laser range finders are suitable for the action recognition related task. Such a dataset could examine the action recognition for applications in automotive context. For assisted or autonomous driving, the anticipation of actions becomes especially interesting. Existing datasets, including laser range data, focus on semantic segmentation tasks. Action recognition and related topics like the temporal segmentation, spatio-temporal localization or anticipation remain promising topics for future research. Our imitation learning benchmark could potentially guide future imitation learning approaches and their evaluations to be more reproducible and comparable while still operating on real sensor data.

Appendix A

Simulate Sequence Examples

Trajectories	Initial Frame	Sequence Name
		rect
		circle
		zickzack



heart

Table A.1: Basic motions examples

Trajectories	Initial Frame	Sequence Name
		pour_vispad_into_steka
		reach_bittermandel
		place_iron

		pick_bittermandel
		push_vispad
		move_steka

Table A.2: Motions examples

Trajectories	Initial Frame	Sequence Name
		stack_glis

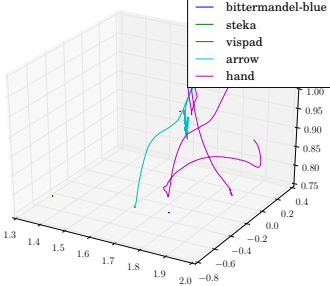
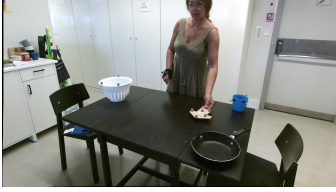
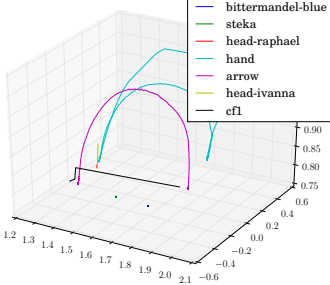
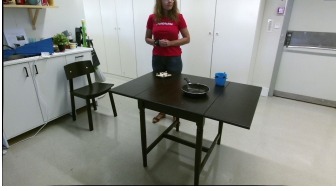
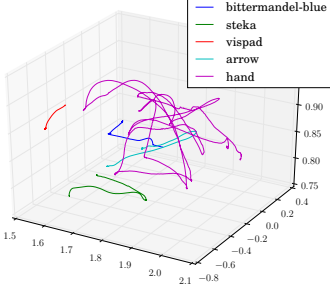
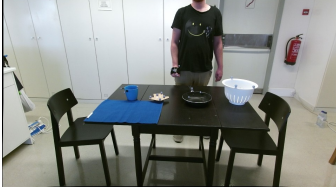
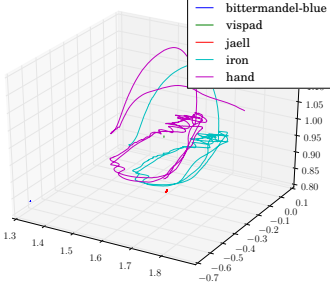

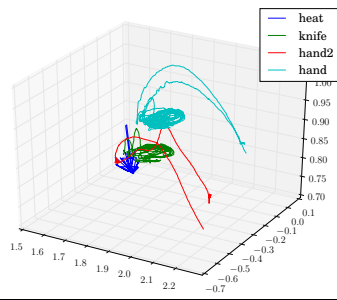
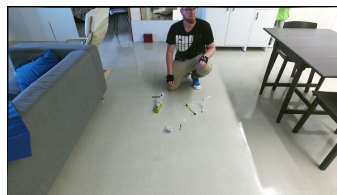
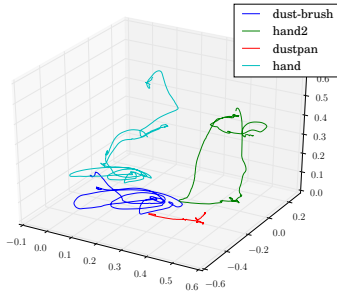
		place_arrow_into_vispad
		pick_and_place_arrow
		rearrange_2018-09-15-17-10-24

Table A.3: Sequential examples

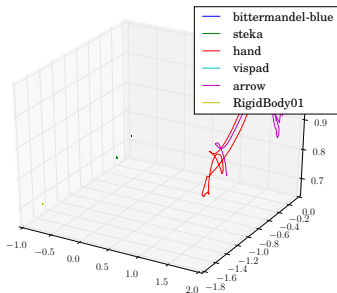
Trajectories	Initial Frame	Sequence Name
		iron_without_cloth



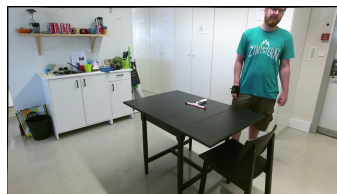
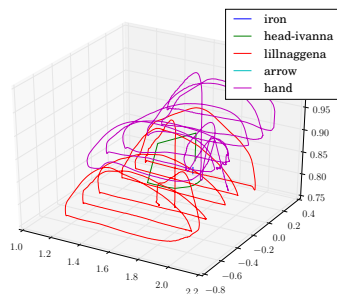
cut



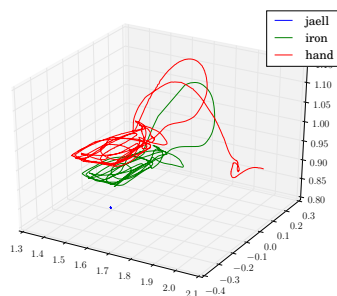
clean_dust



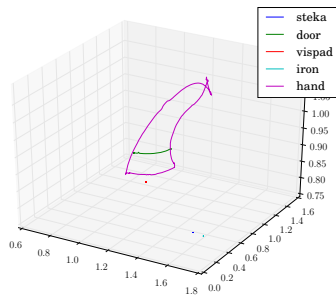
throw_arrow_into_vispad



wipe_table



iron_with_cloth



open_left_cupboard_door

Table A.4: Complex examples

Appendix B

Simitate Testbed Layout

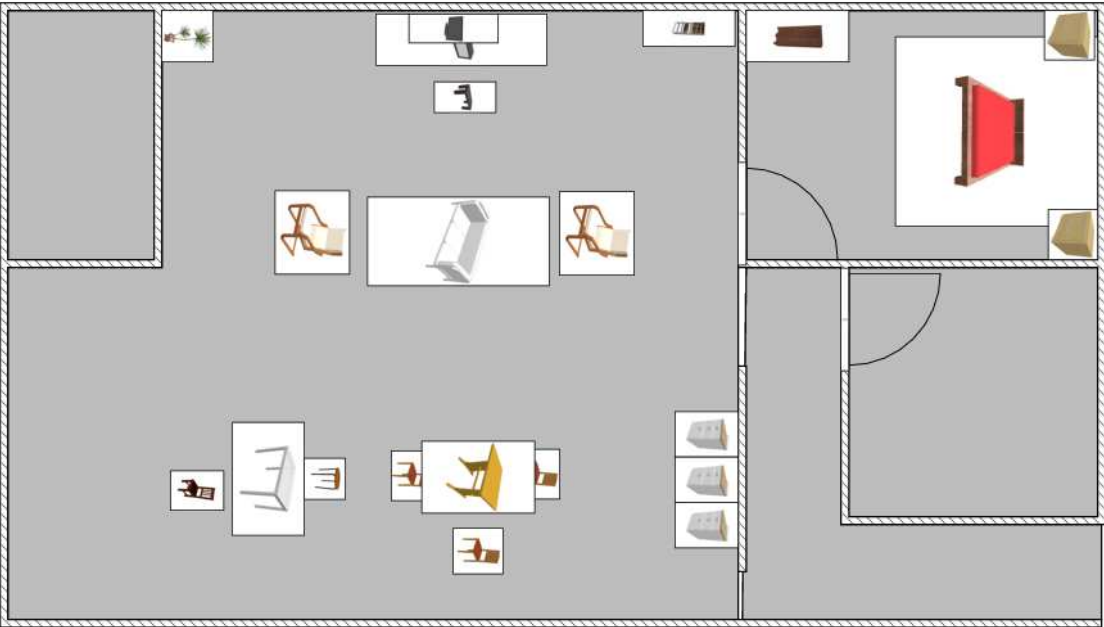


Figure B.1: Simitate Testbed Layout

Appendix C

Curriculum Vitae

- Raphael Memmesheimer
- Emser Str. 104
- 56076 Koblenz

Education

- 2016 – Present, *University of Koblenz-Landau* – Koblenz, Germany, PhD Candidate in Computer Vision, Mentors: Prof. Dr. Dietrich Paulus.
- 2015 – 2016, *University of Koblenz-Landau* – Koblenz, Germany, M.Sc. in Computational Visualistics, Mentors: Prof. Dr. Dietrich Paulus, Dipl. Inform. Nicolai Wojke.
- 2010 – 2015 *University of Koblenz-Landau* – Koblenz, Germany B.Sc. in Computational Visualistics, Mentors: Prof. Dr. Dietrich Paulus, Dipl. Inform. Viktor Seib.

Reviewing Activities

- IROS (IEEE/RSJ International Conference on Intelligent Robots and Systems) in 2020, 2021
- ICRA (IEEE International Conference on Robotics and Automation) in 2020, 2021
- RO-MAN (IEEE International Conference on Robot and Human Interactive Communication) in 2019
- RA-L (IEEE Robotics and Automation Letters) in 2019

Services

- 2019 / 2020 University of Koblenz-Landau *Member for appointments committee, Institute for Computational Visualistics*
- 2021 – 2022 Institute for Computational Visualistics, University of Koblenz-Landau *Marketing*
- 2016 – 2022 Institute for Computational Visualistics, University of Koblenz-Landau *Teaching staff representative for head of institute sessions*
- 2018 – 2020 RoboCup German Open (Magdeburg) *League Co-Chair (RoboCup@Home), together with PD Dr.-Ing. Sven Wachsmuth*

List of Invited Talks

- 2022 Servicerobotik - Wettbewerbe und Anwendungen *MN-Seminar, Hochschule Darmstadt (virtual), Germany*
- 2020 Towards adaptive learning in robot competitions *European Robotics Forum, Malaga, Spain*
- 2020 Roboter und künstliche Intelligenz *Guestlecture: Intelligenz, Denken und Problemlösen / Roboter - KI oder Bauplan für eine Seele, Koblenz, Germany*
- 2019 Künstlichen Intelligenz in Haushalt- und Servicerobotik *VDI Mittelrhein Jahresversammlung, Koblenz, Germany*
- 2018 homer@Uni Koblenz *Tsotsos Lab, Toronto, Canada*
- 2018 Lernen durch Demonstration *Guestlecture: Intelligenz, Denken und Problemlösen, Koblenz, Germany*
- 2017 RoboCup 2017 / Where is HRI and where can it go to in RoboCup? RO-MAN Workshop: HRI for Service Robots in RoboCup@Home *IEEE International Symposium on Robot and Human Interactive Communication, Lisbon, Portugal*
- 2017 Report from ERL Service Robots RO-MAN Workshop: HRI for Service Robots in RoboCup@Home *IEEE International Symposium on Robot and Human Interactive Communication, Lisbon, Portugal*
- 2017 Team's experience in ERL - Service Robots European robotics competitions and challenges: status quo and lessons learned *European Robotics Forum, Edinburgh, Scotland*

- 2017 Navigation and Mapping of Team homer@UniKoblenz *European Robotics League, Peccioli, Italy*

Awards

- 2021 Placed 3rd in the Customer Interaction Category of the Future Convenience Store Challenge (private attendance) *World Robot Summit, Nagoya, Japan.*
- 2021 Limbo (private built robot was awarded) *Region56+ Award, Koblenz, Germany.*
- 2021 Placed 1st in the Heart-Met Gesture Recognition Challenge / Placed 1st in the Heart-Met Action Recognition Challenge (AcRec@UniKoblenz) *Metrics Project, Online.*
- 2019 Placed 1st in @Home Open Platform League (World Champion) *RoboCup World Cup, Sydney, Australia.*
- 2019 Best in Professional Service Robots League *European Robotics League, Bonn (Germany), Koblenz (Germany).*
- 2019 Best in Consumer Service Robots League *European Robotics League, Lisbon, Portugal.*
- 2019 Placed 3rd in @Home League *RoboCup German Open, Magdeburg.*
- 2018 Placed 1st in @Home Open Platform League (World Champion) / Best Poster Award Open Platform League *RoboCup World Cup, Montreal, Canada.*
- 2018 Placed 3rd in the Customer Interaction Category of the Future Convenience Store Challenge *World Robot Summit, Tokyo, Japan.*
- 2018 (some of them shared equally with other teams) Best in TBM1: Getting to know my home / Best in TBM2: Welcoming Visitors / Best in TBM3: Catering for Grannie Annie's Comfort / Best in TBM5: GPSR *European Robotics League, Lisbon (Portugal), Edinburgh (Scotland), Barcelona (Spain).*
- 2018 Placed 1st in @Home League *RoboCup German Open, Magdeburg.*
- 2017 Placed 1st in @Home Open Platform League (World Champion) *RoboCup World Cup, Nagoya, Japan.*
- 2017 Lehrpreis Sommersemester 2017 | Projekt- und Forschungspraktikum: Robbie *Lehrpreis der Hochschuldidaktischen Arbeitsstelle, Koblenz, Germany.*

- 2017 Finalist *ICRA 2017 DJI RoboMaster Mobile Manipulation Challenge, Singapore, Singapore.*
- 2017 Placed 1st in @Home League *RoboCup German Open, Magdeburg.*
- 2017 Placed 1st in the chair valuation *Day of Computervisualistics, University of Koblenz.*
- 2017 Placed 2nd *DJI RoboMaster Technical Challenge, Shenzhen, China.*
- 2016 Placed 2nd in @Home League *RoboCup European Open, Eindhoven, Netherlands.*
- 2016 Finalist in @Home League *RoboCup World Cup, Leipzig, Germany.*
- 2015 Placed 1st in @Home League (World Champion) Placed 1st in Speech Recognition and Audio Detection Best Looking Robot Award *RoboCup World Cup, Hefei, China.*
- 2015 1st in overall ranking (together with team SocRob) Best Team award *RoCKIn, Lisboa, Portugal.*
- 2015 Best Demonstration in RoCKIn@Home track *RoCKIn Camp, Peccioli, Italy.*
- 2015 Placed 2nd in @Home League *RoboCup German Open, Magdeburg.*
- 2015 Placed 1st in the chair valuation *Day of Computervisualistics, University of Koblenz.*
- 2014 1st in the @Home Track 2nd in Object Recognition *RoCKIn, Toulouse, France.*
- 2014 Finalist in the @Home League *RoboCup German Open, Magdeburg.*
- 2014 Best Final Demonstration in the RoCKIn@Home track *RoCKIn Camp, Rome.*
- 2013 Placed 3rd in the chair valuation Placed 1st in the audience valuation *Day of Computervisualistics, University of Koblenz.*
- 2013 Finalist in the @Home League *RoboCup WorldCup, Eindhoven, Netherlands.*
- 2013 Placed 3rd in the @Home League *RoboCup German Open, Magdeburg.*

Supervised Theses

1. *Niko Schmidt, B.Sc.* (2017) Evaluation von physiotherapeutischen Übungen durch Analyse von Tiefenbildern
2. *Lukas Buchhold, B.Sc.* (2018) Simultaneous Object Localization and Classification based on Semantic Segmentation
3. *Nick Theisen, M.Sc.* (2018) Inverse Reinforcement Learning for Robotic Manipulation Tasks
4. *Simon Häring, B.Sc.* (2018) Object detection using convolutional neural networks
5. *Gregor Heuer, M.Sc.* (2018) Markerless Action Recognition
6. *Jannis Eisenmenger, B.Sc.* (2019) Reinforcement Learning for Robot Manipulation Tasks
7. *Jasvinder Kaur, M.Sc.* (2019) Detection of Human-Object Interactions
8. *Thies Möhlenhof, M.Sc.* (2019) Visual Object Tracking and Prediction Using Recurrent Neural Networks
9. *Lukas Debold, M.Sc.* (2019) Simultaneous Action and Object Classification in Videos using Convolutional Neural Networks
10. *Ivanna Kramer, M.Sc.* (2019) Imitating Manipulation Tasks Using Generative Adversarial Networks
11. *Tobias Evers, B.Sc.* (2019) Adaptives Lernen durch Demonstration mittels Objektinteraktion
12. *Robin Bartsch, B.Sc.* (2019) Generative Adversarial Networks for Image Data Augmentation
13. *Alex Weissörtel, B.Sc.* (2020) Evaluation verschiedener Netzwerkarchitekturen für Aktionserkennung auf Signalrepräsentationen
14. *Simon Häring, M.Sc.* (2020) Action Segmentation on Representations of Skeleton Sequences using Transformer Networks
15. *Michael Duhme, M.Sc.* (2020) Multimodal Action Recognition using Graph Convolutional Neural Networks
16. *Ida Germann, B.Sc.* (2021) Action Recognition on Skeleton Sequences Using Graph Convolutional Neural Networks

17. *Katrina Mahlendorf, B.Sc.* (2022) Action Recognition on Egocentric Video Sequences in Natural Environments
18. *Matthias Wellstein, B.Sc.* (2022) Sign language recognition using convolutional neural networks

List of Tables

3.1	Modality support comparison of various approaches	35
3.2	Action recognition datasets used in the experiments	47
3.3	Results on NTU RGB+D 120. Units are in %.	51
3.4	Results on UTD-MHAD. Units are in %.	52
3.5	Results on ARIL dataset. Units are in %.	54
3.6	Results on Simitate. Units are in %.	55
3.7	ETRI-Activity-3D action recognition results	56
3.8	Toyota smarhome results	57
3.9	UAV-human results	59
3.10	Kinetics-400 skeleton results	59
3.11	Action recognition results on various datasets	61
4.1	Comparison to the State-of-the-Art on the UTD-MHAD dataset	75
4.2	Comparison to the State-of-the-Art on the MMAct dataset	76
4.3	Top 5 improved classes by the fusion	79
5.1	Results for different auxiliary training set sizes for one-shot recognition on the NTU RGB+D 120 dataset.	101
5.2	Ablation study for our proposed one-shot action recognition approach on the NTU RGB+D 120 dataset.	101
5.3	One-shot action recognition results on the UTD-MHAD dataset.	102
5.4	Inter-joint one-shot action recognition results on the UTD-MHAD dataset.	103
5.5	Inter-modal one-shot action recognition results	103
5.6	One-shot action recognition results on the Simitate dataset.	104
5.7	One-shot action recognition results on the <i>NTU RGB+D 120</i> dataset.	107
5.8	Results for different auxiliary training set sizes for one-shot recognition on the <i>NTU RGB+D 120</i> dataset in %.	107
5.9	Skeleton-DML ablation study	109
5.10	Ablation study for different representations.	110

6.1	Results for dense representations using different coordinate encoding methods.	122
6.2	Results with the standard Kinect v2 and the TSSI joint orders and varying normalization methods.	122
6.3	Ablation study modal feature	124
6.4	Results with two skeletons	124
6.5	Comparison of dense- and sparse representations.	125
6.6	Comparison with related approaches.	126
7.1	Simulate object list	142
7.2	Sequence overview	143
7.3	Evaluation of the absolute translation error (units are in m)	148
A.1	Basic motions examples	156
A.2	Motions examples	157
A.3	Sequential examples	158
A.4	Complex examples	160

List of Figures

1.1	Sensors used for action recognition.	3
1.2	Different setups for action recognition	3
1.3	Action recognition application examples	4
1.4	Imitation learning application examples	4
1.5	Video abstract previews for the core contributions	11
2.1	Classification of action recognition approaches.	15
2.2	IoU visualization	18
2.3	Convolutional Neural Network overview	20
2.4	Example of a convolution operation	21
2.5	Activation functions	21
2.6	A max pool operation with (2,2).	22
2.7	Skeleton Graph Representation for Spatio-Temporal Action Recognition. Copyright © 2019, IEEE	25
2.8	Metric Learning goal	26
2.9	Human pose models, commonly used in human pose estimation.	27
2.10	Depth-based skeleton estimation approach. Copyright © 2011, IEEE	27
2.11	OpenPose pose estimation approach. Copyright © 2021, IEEE	28
2.12	Uni-modal architecture	29
2.13	Early Fusion and Late Fusion architectures	30
2.14	Cross-Modal architecture	30
2.15	Inter-Modal architecture	31
3.1	Representation overview for various sensor modalities	34
3.2	Action recognition approach overview	40
3.3	Sparse sample representations for skeleton and IMU data	42
3.4	Sparse sample representations for Wi-Fi CSI fingerprints	43
3.5	Dense representation examples	43
3.6	Cross setup protocol	46
3.7	Cross subject protocol	46
3.8	NTU RGBD 120 confusion matrices	50

3.9	UTD-MHAD confusion matrices	52
3.10	ARIL confusion matrix	53
3.11	Simitate confusion matrix	54
3.12	ETRI-Activity-3D confusion matrix	56
3.13	Toyota Smarthome, Dense Representation, 150 epochs.	57
3.14	UAV-Human confusion matrix	58
3.15	Kinetics-400 confusion matrix	60
4.1	Skeleton + IMU Graph Nodes Variations	66
4.2	Fusion of Skeleton and Wearable Sensor Signals	69
4.3	Combined Early Fusion Approaches	72
4.4	Skeleton + RGB Fusion Models	77
4.5	Confusion matrix for the MMAct dataset	78
4.6	Class specific accuracies for all MMAct classes	79
5.1	Illustrative example for an application of our one-shot action recognition approach	85
5.2	One-Shot action recognition approach overview	90
5.3	Exemplary representation for a throwing activity of the NTU-RGB+D 120 dataset.	91
5.4	Illustrative example for an application of the Skeleton-DML approach	94
5.5	Skeleton-DML representation construction	95
5.6	Skeleton-DML representation example	96
5.7	Metric learning training overview	96
5.8	One-Shot action recognition results with SL-DML on the NTU RGB+D 120 dataset	100
5.9	UMAP embedding visualization for the one-shot experiments	102
5.10	Inter-joint and inter-modal experiment setup	104
5.11	UMAP embedding visualization for the inter-modal experiments	105
5.12	Result graph for increasing auxiliary set sizes.	106
5.13	Skeleton-DML representation examples	108
5.14	UMAP embedding visualization for Skeleton-DML on the NTU 120 one-shot protocol	112
6.1	Action segmentation approach overview	116
6.2	Action segmentation overview with DETR	118
6.3	Construction of a Representation	121
6.4	Action segmentation per class results on the PKU-MMD dataset	123
6.5	Example segmentation results for the PKU-MMD dataset	126
6.6	Exemplary action segmentation results on the PKU-MMD dataset	127
7.1	Simitate benchmarking overview	131

7.2	Dataset setup	134
7.3	ISRoboNet@Home testbed	135
7.4	Checkerboard with reflective markers	136
7.5	Extrinsic transformation overview	137
7.6	Motion Capture / RGB+D Camera Calibration setup	138
7.7	Visualization of the extrinsic calibration results	140
7.8	Objects used for the dataset.	141
7.9	Example sequences	144
7.10	Effect metric visualization	145
7.11	Trajectory metric visualization	146
7.12	Trajectory comparison in simulation	147
B.1	Simulate Testbed Layout	161

Acronyms

ADAM Adaptive Moment Estimation. 153

APE Absolute Pose Error. 153

CA Cosine Annealing. 153

CAWR Cosine Annealing with Warm Restarts. 153

CNN Convolutional Neural Network. 7, 12, 13, 19–22, 24, 28, 33, 35–39, 41, 44, 47, 50, 51, 53, 58–60, 62, 65, 66, 68, 71, 74, 92, 119, 149, 150, 153

CSI Channel State Information. 2, 34, 43, 45, 49, 53, 54, 149, 153, 171

DETR DEtection TRansformer. ix, 115, 116, 118, 119, 121, 122, 128, 150, 153, 172

GCN Graph Convolutional Network. 7, 12, 13, 24, 26, 33, 38, 47, 51, 59, 60, 63, 65–67, 69–72, 74, 80, 81, 149, 150, 153

GFT Graph Fourier Transform. 153

GIoU Generalized Intersection over Union. 18, 115, 121, 125, 153

GNN Graph Neural Network. 153

GPS Global Positioning System. 2, 153

GRU Gated Recurrent Unit. 153

HAR Human Action Recognition. 65–67, 74, 153

HMM Hidden Markov Model. 117, 153

HRI Human Robot Interaction. 3, 153

IDT Improved Dense Trajectories. 117, 153

- IMU** Inertial Measurement Unit. 2, 29–31, 35, 42, 43, 45, 47, 53, 62, 65, 69–74, 76–78, 80, 86, 99, 100, 102, 105, 113, 149, 153, 171
- IoU** Intersection over Union. 18, 153, 171
- IR** infrared. 27, 47, 153
- LSTM** Long Short-Term Memory. 37, 38, 50, 51, 57, 59, 89, 118, 153
- mAP** Mean Average Precision. 18, 19, 121, 122, 124–126, 153
- MHI** Motion History Image. 153
- MLP** Multilayer Perceptron. 77, 153
- mpcA** Mean Per-Class Accuracy. 17, 55, 57, 153
- NAS** Neural Architecture Search. 67, 129, 151, 153
- PCA** Principal Component Analysis. 117, 153
- ReLU** Rectified Linear Unit. 22, 24, 153
- RGB** Red Green Blue. 13, 26–28, 30, 31, 38, 73–77, 80, 136, 137, 149, 153, 172, 173
- RGB-D** Red Green Blue Depth. 13, 26, 27, 48, 130, 134, 136–139, 151, 153
- RMSE** Root Mean Square Error. 146, 153
- RNN** Recurrent Neural Network. 38, 51, 118, 153
- RPE** Relative Pose Error. 144, 153
- SDG** Stochastic Gradient Decent. 153
- SGD** Stochastic Gradient Descent. 24, 153
- ST-GCN** Spatial Temporal Graph Convolutional Network. 25, 51, 58–60, 153
- SVM** Support Vector Machine. 37, 153
- TCN** Temporal Convolutional Network. 117, 153
- TSN** Temporal Segment Network. 117, 153
- UAV** Unmanned Aerial Vehicle. 2, 3, 45, 153

Publications

Thesis Related

- [DMP21] Michael Duhme, **Raphael Memmesheimer**, and Dietrich Paulus. “Fusion-GCN: Multimodal Action Recognition Using Graph Convolutional Networks”. In: *Pattern Recognition - 43rd DAGM German Conference, DAGM GCPR 2021, Bonn, Germany, September 28 - October 1, 2021, Proceedings*. Ed. by Christian Bauckhage, Juergen Gall, and Alexander G. Schwing. Vol. 13024. Lecture Notes in Computer Science. Springer, 2021, pp. 265–281. DOI: 10.1007/978-3-030-92659-5_17.
- [HMP21] Simon Häring, **Raphael Memmesheimer**, and Dietrich Paulus. “Action Segmentation on Representations of Skeleton Sequences Using Transformer Networks”. In: *2021 IEEE International Conference on Image Processing, ICIP 2021, Anchorage, AK, USA, September 19-22, 2021*. IEEE, 2021, pp. 3053–3057. DOI: 10.1109/ICIP42928.2021.9506687.
- [Mem+19] **Raphael Memmesheimer**, Ivanna Kramer, Viktor Seib, and Dietrich Paulus. “Simitate: A Hybrid Imitation Learning Benchmark”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2019, Macau, SAR, China, November 3-8, 2019*. IEEE, 2019, pp. 5243–5249. DOI: 10.1109/IROS40897.2019.8968029.
- [Mem+22] **Raphael Memmesheimer**, Simon Häring, Nick Theisen, and Dietrich Paulus. “Skeleton-DML: Deep Metric Learning for Skeleton-Based One-Shot Action Recognition”. In: *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2022, Waikoloa, HI, USA, January 3-8, 2022*. IEEE, 2022, pp. 837–845. DOI: 10.1109/WACV51458.2022.00091.
- [MTP20a] **Raphael Memmesheimer**, Nick Theisen, and Dietrich Paulus. “Gimme Signals: Discriminative signal encoding for multimodal activity recognition”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January*

24, 2021. IEEE, 2020, pp. 10394–10401. DOI: 10.1109/IROS45743.2020.9341699.

- [MTP20b] **Raphael Memmesheimer**, Nick Theisen, and Dietrich Paulus. “SL-DML: Signal Level Deep Metric Learning for Multimodal One-Shot Action Recognition”. In: *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*. IEEE, 2020, pp. 4573–4580. DOI: 10.1109/ICPR48806.2021.9413336.

Further Publications

- [KMP21] Ivanna Kramer, **Raphael Memmesheimer**, and Dietrich Paulus. “Customer Interaction of a Future Convenience Store with a Mobile Manipulation Service Robot”. In: *2021 IEEE International Conference on Omni-Layer Intelligent Systems, COINS 2021, Barcelona, Spain, August 23-25, 2021*. IEEE, 2021, pp. 1–7. DOI: 10.1109/COINS51742.2021.9524229.
- [Kra+19] Ivanna Kramer, Niko Schmidt, **Raphael Memmesheimer**, and Dietrich Paulus. “Evaluation Of Physical Therapy Through Analysis Of Depth Images”. In: *28th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2019, New Delhi, India, October 14-18, 2019*. IEEE, 2019, pp. 1–6. DOI: 10.1109/RO-MAN46459.2019.8956435.
- [Mat+18] Mauricio Matamoros, Viktor Seib, **Raphael Memmesheimer**, and Dietrich Paulus. “RoboCup@Home: Summarizing achievements in over eleven years of competition”. In: *2018 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2018, Torres Vedras, Portugal, April 25-27, 2018*. Ed. by Hugo Costelha, João M. F. Calado, Luís Conde Bento, Nuno Lopes, and Paulo Oliveira. IEEE, 2018, pp. 186–191. DOI: 10.1109/ICARSC.2018.8374181.
- [Mem+18] **Raphael Memmesheimer**, Ivanna Mykhalchyshyna, Viktor Seib, Tobias Evers, and Dietrich Paulus. “homer@UniKoblenz: Winning Team of the RoboCup@Home Open Platform League 2018”. In: *RoboCup 2018: Robot World Cup XXII [Montreal, QC, Canada, June 18-22, 2018]*. Ed. by Dirk Holz, Katie Genter, Maarouf Saad, and Oskar von Stryk. Vol. 11374. Lecture Notes in Computer Science. Springer, 2018, pp. 512–523. DOI: 10.1007/978-3-030-27544-0_42.

- [Mem+19a] **Raphael Memmesheimer**, Isabelle Kuhlmann, Mark Mints, Patrik Schmidt, Christian Korbach, Ida Germann, and Dietrich Paulus. “Scratchy: A Lightweight Modular Autonomous Robot for Robotic Competitions”. In: *2019 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2019, Porto, Portugal, April 24-26, 2019*. Ed. by Luís Almeida, Luís Paulo Reis, and António Paulo Moreira. IEEE, 2019, pp. 1–6. DOI: 10.1109/ICARSC.2019.8733655.
- [Mem+19b] **Raphael Memmesheimer**, Viktor Seib, Tobias Evers, Daniel Müller, and Dietrich Paulus. “Adaptive Learning Methods for Autonomous Mobile Manipulation in RoboCup@Home”. In: *RoboCup 2019: Robot World Cup XXIII [Sydney, NSW, Australia, July 8, 2019]*. Ed. by Stephan K. Chalup, Tim Niemüller, Jackrit Suthakorn, and Mary-Anne Williams. Vol. 11531. Lecture Notes in Computer Science. Springer, 2019, pp. 565–577. DOI: 10.1007/978-3-030-35699-6_46.
- [Mem+20] **Raphael Memmesheimer**, Ivanna Kramer, Viktor Seib, Nick Theisen, and Dietrich Paulus. “Robotic Imitation by Markerless Visual Observation and Semantic Associations”. In: *2020 IEEE International Conference on Autonomous Robot Systems and Competitions, ICARSC 2020, Ponta Delgada, Portugal, April 15-17, 2020*. IEEE, 2020, pp. 275–280. DOI: 10.1109/ICARSC49921.2020.9096123.
- [MMP18] **Raphael Memmesheimer**, Ivanna Mykhalchyshyna, and Dietrich Paulus. “Gesture Recognition On Human Pose Features Of Single Images”. In: *9th IEEE International Conference on Intelligent Systems, IS 2018, Funchal, Madeira, Portugal, September 25-27, 2018*. Ed. by Ricardo Jardim-Gonçalves, João Pedro Mendonça, Vladimir Jotsov, Maria Marques, João Martins, and Robert E. Bierwolf. IEEE, 2018, pp. 813–819. DOI: 10.1109/IS.2018.8710515.
- [MSP17] **Raphael Memmesheimer**, Viktor Seib, and Dietrich Paulus. “homer@UniKoblenz: Winning Team of the RoboCup@Home Open Platform League 2017”. In: *RoboCup 2017: Robot World Cup XXI [Nagoya, Japan, July 27-31, 2017]*. Ed. by Hidehisa Akiyama, Oliver Obst, Claude Sammut, and Flavio Tonidandel. Vol. 11175. Lecture Notes in Computer Science. Springer, 2017, pp. 509–520. DOI: 10.1007/978-3-030-00308-1_42.
- [Roa+21] Máximo A. Roa, Mehmet Remzi Dogar, Jordi Pagès, Carlos Vivas, Antonio Morales, Nikolaus Correll, Michael Gerner, Jan Rosell, Sergi Foix, **Raphael Memmesheimer**, and Francesco Ferro. “Mobile Manipulation Hackathon: Moving into Real World Applications”. In: *IEEE Robotics*

- Autom. Mag.* 28.2 (2021), pp. 112–124. DOI: 10.1109/MRA.2021.3061951.
- [Sch+19] Pascal Schneider, **Raphael Memmesheimer**, Ivanna Kramer, and Dietrich Paulus. “Gesture Recognition in RGB Videos Using Human Body Keypoints and Dynamic Time Warping”. In: *RoboCup 2019: Robot World Cup XXIII [Sydney, NSW, Australia, July 8, 2019]*. Ed. by Stephan K. Chalup, Tim Niemüller, Jackrit Suthakorn, and Mary-Anne Williams. Vol. 11531. Lecture Notes in Computer Science. Springer, 2019, pp. 281–293. DOI: 10.1007/978-3-030-35699-6_22.
- [Sei+15] Viktor Seib, Stephan Manthe, **Raphael Memmesheimer**, Florian Polster, and Dietrich Paulus. “Team Homer@UniKoblenz - Approaches and Contributions to the RoboCup@Home Competition”. In: *RoboCup 2015: Robot World Cup XIX [papers from the 19th Annual RoboCup International Symposium, Hefei, China, July 23, 2015]*. Ed. by Luís Almeida, Jianmin Ji, Gerald Steinbauer, and Sean Luke. Vol. 9513. Lecture Notes in Computer Science. Springer, 2015, pp. 83–94. DOI: 10.1007/978-3-319-29339-4_7.
- [SMP15] Viktor Seib, **Raphael Memmesheimer**, and Dietrich Paulus. “Ensemble classifier for joint object instance and category recognition on RGB-D data”. In: *2015 IEEE International Conference on Image Processing, ICIP 2015, Quebec City, QC, Canada, September 27-30, 2015*. IEEE, 2015, pp. 143–147. DOI: 10.1109/ICIP.2015.7350776.
- [SMP16] Viktor Seib, Raphael Memmesheimer, and Dietrich Paulus. “A ROS-based System for an Autonomous Service Robot”. In: *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Ed. by Anis Koubaa. Vol. 625. Studies in Computational Intelligence. Heidelberg: Springer, Mar. 2016, pp. 215–252. ISBN: 978-3-319-26054-9.
- [Van+17] Andrea Vanzo, Luca Iocchi, Daniele Nardi, **Raphael Memmesheimer**, Dietrich Paulus, Iryna Ivanovska, and Gerhard K. Kraetzschmar. “Benchmarking Speech Understanding in Service Robotics”. In: *Proceedings of the 4th Italian Workshop on Artificial Intelligence and Robotics A workshop of the XVI International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017), Bari, Italy, November 14-15, 2017*. Ed. by Salvatore Maria Anzalone, Alessandro Farinelli, Alberto Finzi, and Fulvio Mastrogiovanni. Vol. 2054. CEUR Workshop Proceedings. CEUR-WS.org, 2017, pp. 34–40.
- [WMP17] Nicolai Wojke, **Raphael Memmesheimer**, and Dietrich Paulus. “Joint operator detection and tracking for person following from mobile platforms”. In: *20th International Conference on Information Fusion, FU-*

SION 2017, Xi'an, China, July 10-13, 2017. IEEE, 2017, pp. 1–8. DOI: 10.23919/ICIF.2017.8009746.

Technical Reports

- [Mem+16] **Raphael Memmesheimer**, Viktor Seib, Gregor Heuer, Patrik Schmidt, Darius Thies, Ivanna Mykhalchyshyna, Johannes Klöckner, Martin Schmitz, Niklas Yann Wettengel, Nils Geilen, Richard Schütz, Florian Polster, and Dietrich Paulus. *RoboCup 2016 - homer@UniKoblenz (Germany)*. Tech. rep. 1/2018. University of Koblenz-Landau, 2016.
- [Mem+18a] **Raphael Memmesheimer**, Viktor Seib, Niklas Yann Wettengel, Florian Polster, Daniel Müller, Moritz Löhne, Malte Roosen, Ivanna Mykhalchyshyna, Lukas Buchhold, Matthias Schnorr, and Dietrich Paulus. *RoboCup 2017 - homer@UniKoblenz (Germany)*. Tech. rep. 2/2018. Koblenz, Germany: Universität Koblenz-Landau, Fachbereich Informatik, 2018.
- [Mem+18b] **Raphael Memmesheimer**, Niklas Yann Wettengel, Lukas Debald, Anatoli Eckert, Thies Möhlenhof, Tobias Evers, Gregor Heuer, Nick Theisen, Lukas Buchhold, Jannis Eisenmenger, Simon Häring, and Dietrich Paulus. *RoboCup 2018 - homer@UniKoblenz (Germany)*. Tech. rep. 4/2018. Universität Koblenz-Landau, Fachbereich Informatik, 2018.
- [Mem+19] **Raphael Memmesheimer**, Daniel Müller, Ivanna Kramer, Niklas Yann Wettengel, Tobias Evers, Lukas Buchhold, Patrik Schmidt, Niko Schmidt, Ida Germann, Mark Mints, Greta Rettler, Christian Korbach, Robin Bartsch, Kuhlmann Isabelle, Thomas Weiland, and Dietrich Paulus. *RoboCup@Home Open Platform League 2019 - homer@UniKoblenz (Germany)*. Tech. rep. 1/2019. Koblenz, Germany: Universität Koblenz-Landau, Fachbereich Informatik, 2019.

Bibliography

- [AC99] Jake K. Aggarwal and Quin Cai. “Human Motion Analysis: A Review”. In: *Comput. Vis. Image Underst.* 73.3 (1999), pp. 428–440. DOI: 10.1006/cviu.1998.0744.
- [AEP06] Andreas Argyriou, Theodoros Evgeniou, and Massimiliano Pontil. “Multi-Task Feature Learning”. In: *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*. Ed. by Bernhard Schölkopf, John C. Platt, and Thomas Hofmann. MIT Press, 2006, pp. 41–48. DOI: <https://doi.org/10.7551/mitpress/7503.001.0001>.
- [AH15] Sharath Chandra Akkaladevi and Christoph Heindl. “Action recognition for human robot interaction in industrial applications”. In: *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)*. IEEE, 2015, pp. 94–99. DOI: 10.1109/CGVIS.2015.7449900.
- [Al+18] Mohammad Al-Naser, Hiroki Ohashi, Sheraz Ahmed, Katsuyuki Nakamura, Takayuki Akiyama, Takuto Sato, Phong Xuan Nguyen, and Andreas Dengel. “Hierarchical Model for Zero-shot Activity Recognition using Wearable Sensors”. In: *Proceedings of the 10th International Conference on Agents and Artificial Intelligence, ICAART 2018, Volume 2, Funchal, Madeira, Portugal, January 16-18, 2018*. Ed. by Ana Paula Rocha and H. Jaap van den Herik. SciTePress, 2018, pp. 478–485. DOI: 10.5220/0006595204780485.
- [APB17] Kamel Abdelouahab, Maxime Pelcat, and François Berry. “Why TanH is a Hardware Friendly Activation Function for CNNs”. In: *Proceedings of the 11th International Conference on Distributed Smart Cameras, Stanford, CA, USA, September 5-7, 2017*. Ed. by Miguel O. Arias-Estrada, Christian Micheloni, Hamid K. Aghajan, Octavia I. Camps, and Víctor M. Brea. ACM, 2017, pp. 199–201. DOI: 10.1145/3131885.3131937.

- [AR11] J. K. Aggarwal and M. S. Ryoo. “Human activity analysis: A review”. In: *ACM Comput. Surv.* 43.3 (2011), 16:1–16:43. DOI: 10.1145/1922649.1922653.
- [Arg+09] Brenna D. Argall, Sonia Chernova, Manuela M. Veloso, and Brett Browning. “A survey of robot learning from demonstration”. In: *Robotics Auton. Syst.* 57.5 (2009), pp. 469–483. DOI: 10.1016/j.robot.2008.10.024.
- [Bac+20] Mohamed Hedi Baccour, Frauke Driewer, Tim Schäck, and Enkelejda Kasneci. “Camera-based Driver Drowsiness State Classification Using Logistic Regression Models”. In: *2020 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2020, Toronto, ON, Canada, October 11-14, 2020*. IEEE, 2020, pp. 1–8. DOI: 10.1109/SMC42975.2020.9282918.
- [Bat+17] Tamas Bates, Karinne Ramirez-Amaro, Tetsunari Inamura, and Gordon Cheng. “On-line simultaneous learning and recognition of everyday activities from virtual reality performances”. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*. IEEE, 2017, pp. 3510–3515. DOI: 10.1109/IROS.2017.8206193.
- [BC94] Donald J. Berndt and James Clifford. “Using Dynamic Time Warping to Find Patterns in Time Series”. In: *Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop, Seattle, Washington, USA, July 1994. Technical Report WS-94-03*. Ed. by Usama M. Fayyad and Ramasamy Uthurusamy. AAAI Press, 1994, pp. 359–370.
- [BKC17] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. “SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.12 (2017), pp. 2481–2495. DOI: 10.1109/TPAMI.2016.2644615.
- [Bob97] Aaron F Bobick. “Movement, activity and action: the role of knowledge in the perception of motion”. In: *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences* 352.1358 (1997), pp. 1257–1265.
- [Bra+20] Samarth Brahmhatt, Chengcheng Tang, Christopher D. Twigg, Charles C. Kemp, and James Hays. “ContactPose: A Dataset of Grasps with Object Contact and Hand Pose”. In: *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XIII*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12358. Lecture Notes in Computer Science.

- Springer, 2020, pp. 361–378. DOI: 10.1007/978-3-030-58601-0_22.
- [BS08] Keni Bernardin and Rainer Stiefelhagen. “Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics”. In: *EURASIP J. Image and Video Processing 2008* (2008). DOI: 10.1155/2008/246309.
- [Bui+17] Tu Bui, Leonardo Sampaio Ferraz Ribeiro, Moacir Ponti, and John P. Collomosse. “Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network”. In: *Comput. Vis. Image Underst.* 164 (2017), pp. 27–37. DOI: 10.1016/j.cviu.2017.06.007.
- [Cae+19] Carlos Caetano, Jessica Sena de Souza, François Brémond, Jefersson A. dos Santos, and William Robson Schwartz. “SkeleMotion: A New Representation of Skeleton Joint Sequences based on Motion Information for 3D Action Recognition”. In: *16th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2019, Taipei, Taiwan, September 18-21, 2019*. IEEE, 2019, pp. 1–8. DOI: 10.1109/AVSS.2019.8909840.
- [Cai+21] Jinmiao Cai, Nianjuan Jiang, Xiaoguang Han, Kui Jia, and Jiangbo Lu. “JOLO-GCN: Mining Joint-Centered Light-Weight Information for Skeleton-Based Action Recognition”. In: *IEEE Winter Conference on Applications of Computer Vision, WACV 2021, Waikoloa, HI, USA, January 3-8, 2021*. IEEE, 2021, pp. 2734–2743. DOI: 10.1109/WACV48630.2021.00278.
- [Cao+16] Congqi Cao, Yifan Zhang, Chunjie Zhang, and Hanqing Lu. “Action Recognition with Joints-Pooled 3D Deep Convolutional Descriptors”. In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*. Ed. by Subbarao Kambhampati. IJCAI/AAAI Press, 2016, pp. 3324–3330. ISBN: 9781577357704. DOI: 10.5555/3061053.3061086.
- [Cao+17] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. “Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1302–1310. DOI: 10.1109/CVPR.2017.143.
- [Cao+18] Congqi Cao, Yifan Zhang, Chunjie Zhang, and Hanqing Lu. In: *IEEE Trans. Cybern.* 48.3 (2018), pp. 1095–1108. DOI: 10.1109/TCYB.2017.2756840.

- [Cao+19] Congqi Cao, Cuiling Lan, Yifan Zhang, Wenjun Zeng, Hanqing Lu, and Yanning Zhang. “Skeleton-Based Action Recognition With Gated Convolutional Neural Networks”. In: *IEEE Trans. Circuits Syst. Video Technol.* 29.11 (2019), pp. 3247–3257. DOI: 10.1109/TCSVT.2018.2879913.
- [Cao+21] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. “OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 43.1 (2021), pp. 172–186. DOI: 10.1109/TPAMI.2019.2929257.
- [Car+19] Chris Careaga, Brian Hutchinson, Nathan Oken Hodas, and Lawrence Phillips. “Metric-Based Few-Shot Learning for Video Action Recognition”. In: *CoRR abs/1909.09602* (2019). arXiv: 1909.09602.
- [Car+20] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. “End-to-End Object Detection with Transformers”. In: *CoRR abs/2005.12872* (2020). arXiv: 2005.12872.
- [CB21] Erwin Coumans and Yunfei Bai. *PyBullet, a Python module for physics simulation for games, robotics and machine learning*. <http://pybullet.org>. 2021.
- [CBS19] Carlos Caetano, François Brémond, and William Robson Schwartz. “Skeleton Image Representation for 3D Action Recognition Based on Tree Structure and Reference Joints”. In: *32nd SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2019, Rio de Janeiro, Brazil, October 28-30, 2019*. IEEE, 2019, pp. 16–23. DOI: 10.1109/SIBGRAPI.2019.00011.
- [Cha18] Pragnan Chakravorty. “What Is a Signal? [Lecture Notes]”. In: *IEEE Signal Process. Mag.* 35.5 (2018), pp. 175–177. DOI: 10.1109/MSP.2018.2832195.
- [Che+13] Saisakul Chernbumroong, Shuang Cang, Anthony S. Atkins, and Hongnian Yu. “Elderly activities recognition and classification for applications in assisted living”. In: *Expert Syst. Appl.* 40.5 (2013), pp. 1662–1674. DOI: 10.1016/j.eswa.2012.09.004.
- [Che+20] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. “Skeleton-Based Action Recognition With Shift Graph Convolutional Network”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. IEEE, 2020, pp. 180–189. DOI: 10.1109/CVPR42600.2020.00026.

- [Cho+07] SangJo Choi, JeongHee Kim, DongGu Kwak, Pongtep Angkititrakul, and John HL Hansen. “Analysis and classification of driver behavior using in-vehicle can-bus information”. In: *Biennial workshop on DSP for in-vehicle and mobile systems*. 2007, pp. 17–19.
- [CJK15] Chen Chen, Roozbeh Jafari, and Nasser Kehtarnavaz. “UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor”. In: *2015 IEEE International Conference on Image Processing, ICIP 2015, Quebec City, QC, Canada, September 27-30, 2015*. IEEE, 2015, pp. 168–172. DOI: 10.1109/ICIP.2015.7350781.
- [CKG16a] Konstantinos Charalampous, Ioannis Kostavelis, and Antonios Gasteratos. “Robot navigation in large-scale social maps: An action recognition approach”. In: *Expert Syst. Appl.* 66 (2016), pp. 261–273. DOI: 10.1016/j.eswa.2016.09.026.
- [CKG16b] Konstantinos Charalampous, Ioannis Kostavelis, and Antonios Gasteratos. “Robot navigation in large-scale social maps: An action recognition approach”. In: *Expert Syst. Appl.* 66 (2016), pp. 261–273. DOI: 10.1016/j.eswa.2016.09.026.
- [CLS15] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. “P-CNN: Pose-Based CNN Features for Action Recognition”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 3218–3226. DOI: 10.1109/ICCV.2015.368.
- [Cod+18] Felipe Codevilla, Matthias Müller, Antonio López, Vladlen Koltun, and Alexey Dosovitskiy. “End-to-End Driving Via Conditional Imitation Learning”. In: *2018 IEEE International Conference on Robotics and Automation, ICRA 2018, Brisbane, Australia, May 21-25, 2018*. 2018, pp. 1–9. DOI: 10.1109/ICRA.2018.8460487.
- [Cox58] David R Cox. “The regression analysis of binary sequences”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 20.2 (1958), pp. 215–232.
- [CW08] Ronan Collobert and Jason Weston. “A unified architecture for natural language processing: deep neural networks with multitask learning”. In: *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*. Ed. by William W. Cohen, Andrew McCallum, and Sam T. Roweis. Vol. 307. ACM International Conference Proceeding Series. ACM, 2008, pp. 160–167. DOI: 10.1145/1390156.1390177.

- [CZ17] 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 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 4724–4733. DOI: 10.1109/CVPR.2017.502.
- [Dam+21] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. “The EPIC-KITCHENS Dataset: Collection, Challenges and Baselines”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 43.11 (2021), pp. 4125–4141. DOI: 10.1109/TPAMI.2020.2991965.
- [Das+19] Srijan Das, Rui Dai, Michal Koperski, Luca Minciullo, Lorenzo Garattoni, François Brémond, and Gianpiero Francesca. “Toyota Smarthome: Real-World Activities of Daily Living”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 833–842. DOI: 10.1109/ICCV.2019.00092.
- [Den+09] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*. IEEE Computer Society, 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.
- [DFW15] Yong Du, Yun Fu, and Liang Wang. “Skeleton based action recognition with convolutional neural network”. In: *3rd IAPR Asian Conference on Pattern Recognition, ACPR 2015, Kuala Lumpur, Malaysia, November 3-6, 2015*. IEEE, 2015, pp. 579–583. DOI: 10.1109/ACPR.2015.7486569.
- [DG07] Somayeh Danafar and Niloofar Gheissari. “Action Recognition for Surveillance Applications Using Optic Flow and SVM”. In: *Computer Vision - ACCV 2007, 8th Asian Conference on Computer Vision, Tokyo, Japan, November 18-22, 2007, Proceedings, Part II*. Ed. by Yasushi Yagi, Sing Bing Kang, In-So Kweon, and Hongbin Zha. Vol. 4844. Lecture Notes in Computer Science. Springer, 2007, pp. 457–466. DOI: 10.1007/978-3-540-76390-1_45.
- [Dos+17] Alexey Dosovitskiy, Germán Ros, Felipe Codevilla, Antonio M. López, and Vladlen Koltun. “CARLA: An Open Urban Driving Simulator”. In: *1st Annual Conference on Robot Learning, CoRL 2017, Mountain View,*

- California, USA, November 13-15, 2017, Proceedings*. Vol. 78. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1–16.
- [Dua+17] Yan Duan, Marcin Andrychowicz, Bradly C. Stadie, Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. “One-Shot Imitation Learning”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*. 2017, pp. 1087–1098.
- [Duo+05] Thi V. Duong, Hung Hai Bui, Dinh Q. Phung, and Svetha Venkatesh. “Activity Recognition and Abnormality Detection with the Switching Hidden Semi-Markov Model”. In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*. IEEE Computer Society, 2005, pp. 838–845. DOI: 10.1109/CVPR.2005.61.
- [Eha+19] Muhammad Ehatisham-Ul-Haq, Ali Javed, Muhammad Awais Azam, Hafiz MA Malik, Aun Irtaza, Ik Hyun Lee, and Muhammad Tariq Mahmood. “Robust human activity recognition using multimodal feature-level fusion”. In: *IEEE Access* 7 (2019), pp. 60736–60751.
- [FAL17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. “Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks”. In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1126–1135.
- [Fal19] W.A. et al. Falcon. *PyTorch Lightning*. <https://github.com/PyTorchLightning/pytorch-lightning>. 2019.
- [Fan+13] Sean Ryan Fanello, Ilaria Gori, Giorgio Metta, and Francesca Odone. “One-Shot Learning for Real-Time Action Recognition”. In: *Pattern Recognition and Image Analysis - 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5-7, 2013. Proceedings*. Ed. by João M. Sanches, Luisa Micó, and Jaime S. Cardoso. Vol. 7887. Lecture Notes in Computer Science. Springer, 2013, pp. 31–40. DOI: 10.1007/978-3-642-38628-2_4.
- [Fan+17a] Sean Ryan Fanello, Ilaria Gori, Giorgio Metta, and Francesca Odone. “Keep It Simple and Sparse: Real-Time Action Recognition”. In: *Gesture Recognition*. Ed. by Sergio Escalera, Isabelle Guyon, and Vassilis Athitsos. Springer, 2017, pp. 303–328. DOI: 10.1007/978-3-319-57021-1_10.

- [Fan+17b] Haoshu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. “RMPE: Regional Multi-person Pose Estimation”. In: *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2353–2362. DOI: 10.1109/ICCV.2017.256.
- [Fan+19] Bin Fang, Shidong Jia, Di Guo, Muhua Xu, Shuhuan Wen, and Fuchun Sun. “Survey of imitation learning for robotic manipulation”. In: *Int. J. Intell. Robotics Appl.* 3.4 (2019), pp. 362–369. DOI: 10.1007/s41315-019-00103-5.
- [Fei+19] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. “SlowFast Networks for Video Recognition”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 6201–6210. DOI: 10.1109/ICCV.2019.00630.
- [Fer+15] Basura Fernando, Efstratios Gavves, José Oramas M., Amir Ghodrati, and Tinne Tuytelaars. “Modeling video evolution for action recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 5378–5387. DOI: 10.1109/CVPR.2015.7299176.
- [Fu+15] Zhen-Yong Fu, Tao A. Xiang, Elyor Kodirov, and Shaogang Gong. “Zero-shot object recognition by semantic manifold distance”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 2635–2644. DOI: 10.1109/CVPR.2015.7298879.
- [Gai+15] Tobias Gail, Ramona Hoffmann, Markus Miezal, Gabriele Bleser, and Sigrid Leyendecker. “Towards bridging the gap between motion capturing and biomechanical optimal control simulations”. In: *Thematic Conference on Multibody Dynamics*. 2015.
- [Gao+18] Yongbin Gao, Xuehao Xiang, Naixue Xiong, Bo Huang, Hyo Jong Lee, Rad Alrifai, Xiaoyan Jiang, and Zhijun Fang. “Human Action Monitoring for Healthcare Based on Deep Learning”. In: *IEEE Access* 6 (2018), pp. 52277–52285. DOI: 10.1109/ACCESS.2018.2869790.
- [GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
- [GHM16] Saurabh Gupta, Judy Hoffman, and Jitendra Malik. “Cross Modal Distillation for Supervision Transfer”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June*

- 27-30, 2016. IEEE Computer Society, 2016, pp. 2827–2836. DOI: 10.1109/CVPR.2016.309.
- [GLU12] Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for autonomous driving? The KITTI vision benchmark suite”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*. 2012, pp. 3354–3361. DOI: 10.1109/CVPR.2012.6248074.
- [GM15] Saurabh Gupta and Jitendra Malik. “Visual Semantic Role Labeling”. In: *CoRR abs/1505.04474* (2015). arXiv: 1505.04474.
- [Goh+21] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. “Multimodal neurons in artificial neural networks”. In: *Distill* 6.3 (2021), e30.
- [GP18] Tanmay Gangwani and Jian Peng. “Policy Optimization by Genetic Distillation”. In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.
- [Gu+18] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. “Recent advances in convolutional neural networks”. In: *Pattern Recognition* 77 (2018), pp. 354–377.
- [Guo+18] Xin Guo, Song Wang, Yun Tie, Lin Qi, and Ling Guan. “Joint inter-modal and intramodal correlation preservation for semi-paired learning”. In: *Pattern Recognit.* 81 (2018), pp. 36–49. DOI: 10.1016/j.patcog.2018.03.013.
- [HA15] Elad Hoffer and Nir Ailon. “Deep Metric Learning Using Triplet Network”. In: *Similarity-Based Pattern Recognition - Third International Workshop, SIMBAD 2015, Copenhagen, Denmark, October 12-14, 2015, Proceedings*. Ed. by Aasa Feragen, Marcello Pelillo, and Marco Loog. Vol. 9370. Lecture Notes in Computer Science. Springer, 2015, pp. 84–92. DOI: 10.1007/978-3-319-24261-3_7.
- [Han+18] Yamin Han, Peng Zhang, Tao Zhuo, Wei Huang, and Yanning Zhang. “Going deeper with two-stream ConvNets for action recognition in video surveillance”. In: *Pattern Recognit. Lett.* 107 (2018), pp. 83–90. DOI: 10.1016/j.patrec.2017.08.015.

- [Has+19] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael Black. “Resolving 3D Human Pose Ambiguities With 3D Scene Constraints”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 2282–2292. DOI: 10.1109/ICCV.2019.00237.
- [HBL17] Alexander Hermans, Lucas Beyer, and Bastian Leibe. “In Defense of the Triplet Loss for Person Re-Identification”. In: *CoRR abs/1703.07737* (2017). arXiv: 1703.07737.
- [He+15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 1026–1034. DOI: 10.1109/ICCV.2015.123.
- [He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.
- [He+17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. “Mask R-CNN”. In: *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322.
- [HE16] Jonathan Ho and Stefano Ermon. “Generative Adversarial Imitation Learning”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. Ed. by Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett. 2016, pp. 4565–4573. ISBN: 9781510838819. DOI: 10.5555/3157382.3157608.
- [Hei+15] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. “ActivityNet: A large-scale video benchmark for human activity understanding”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 961–970. DOI: 10.1109/CVPR.2015.7298698.
- [Her+17] Shawn Hershey, Sourish Chaudhuri, Daniel P. W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin W.

- Wilson. “CNN architectures for large-scale audio classification”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. IEEE, 2017, pp. 131–135. DOI: 10.1109/ICASSP.2017.7952132.
- [HGS19] Noureldien Hussein, Efstratios Gavves, and Arnold W. M. Smeulders. “Timeception for Complex Action Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 254–263. DOI: 10.1109/CVPR.2019.00034.
- [Hig+17] Irina Higgins, Arka Pal, Andrei A. Rusu, Loïc Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. “DARLA: Improving Zero-Shot Transfer in Reinforcement Learning”. In: *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*. Ed. by Doina Precup and Yee Whye Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1480–1490.
- [Höf+21] Sebastian Höfer, Kostas E. Bekris, Ankur Handa, Juan Camilo Gamboa, Melissa Mozifian, Florian Golemo, Christopher G. Atkeson, Dieter Fox, Ken Goldberg, John Leonard, C. Karen Liu, Jan Peters, Shuran Song, Peter Welinder, and Martha White. “Sim2Real in Robotics and Automation: Applications and Challenges”. In: *IEEE Trans Autom. Sci. Eng.* 18.2 (2021), pp. 398–400. DOI: 10.1109/TASE.2021.3064065.
- [Hog83] David C. Hogg. “Model-based vision: a program to see a walking person”. In: *Image Vis. Comput.* 1.1 (1983), pp. 5–20. DOI: 10.1016/0262-8856(83)90003-3.
- [Hou+18] Jingyi Hou, Xinxiao Wu, Yuchao Sun, and Yunde Jia. “Content-Attention Representation by Factorized Action-Scene Network for Action Recognition”. In: *IEEE Trans. Multim.* 20.6 (2018), pp. 1537–1547. DOI: 10.1109/TMM.2017.2771462.
- [HS97] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Comput.* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
- [HSR19] Meera Hahn, Andrew Silva, and James M. Rehg. “Action2Vec: A Cross-modal Embedding Approach to Action Learning”. In: *CoRR abs/1901.00484* (2019). arXiv: 1901.00484.

- [Hu+07] Weiming Hu, Dan Xie, Zhouyu Fu, Wenrong Zeng, and Stephen J. Maybank. “Semantic-Based Surveillance Video Retrieval”. In: *IEEE Trans. Image Process.* 16.4 (2007), pp. 1168–1181. DOI: 10.1109/TIP.2006.891352.
- [Hu+19] Jianfang Hu, Wei-Shi Zheng, Lianyang Ma, Gang Wang, Jianhuang Lai, and Jianguo Zhang. “Early Action Prediction by Soft Regression”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 41.11 (2019), pp. 2568–2583. DOI: 10.1109/TPAMI.2018.2863279.
- [Huy+09] H. H. Huynh, J. Meunier, J. Sequeira, and M. Daniel. “Real Time Detection, Tracking and Recognition of Medication Intake”. In: *International Journal of Computer and Information Engineering* 3.12 (2009), pp. 2801–2808. ISSN: eISSN: 1307-6892.
- [HVD15] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. “Distilling the Knowledge in a Neural Network”. In: *CoRR* abs/1503.02531 (2015). arXiv: 1503.02531.
- [II20] Md Mofijul Islam and Tariq Iqbal. “HAMLET: A Hierarchical Multimodal Attention-based Human Activity Recognition Algorithm”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January 24, 2021*. IEEE, 2020, pp. 10285–10292. DOI: 10.1109/IROS45743.2020.9340987.
- [II21] Md Mofijul Islam and Tariq Iqbal. “Multi-GAT: A Graphical Attention-Based Hierarchical Multimodal Representation Learning Approach for Human Activity Recognition”. In: *IEEE Robotics Autom. Lett.* 6.2 (2021), pp. 1729–1736. DOI: 10.1109/LRA.2021.3059624.
- [Iqb+18] Umar Iqbal, Andreas Doering, Hashim Yasin, Björn Krüger, Andreas Weber, and Juergen Gall. “A dual-source approach for 3D human pose estimation from single images”. In: *Comput. Vis. Image Underst.* 172 (2018), pp. 37–49. DOI: 10.1016/j.cviu.2018.03.007.
- [IR20] Javed Imran and Balasubramanian Raman. “Evaluating fusion of RGB-D and inertial sensors for multimodal human action recognition”. In: *J. Ambient Intell. Humaniz. Comput.* 11.1 (2020), pp. 189–208. DOI: 10.1007/s12652-019-01239-9.
- [Jam+20] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. “RLBench: The Robot Learning Benchmark & Learning Environment”. In: *IEEE Robotics Autom. Lett.* 5.2 (2020), pp. 3019–3026. DOI: 10.1109/LRA.2020.2974707.

- [Jan+20] Jinhyeok Jang, Dohyung Kim, Cheonshu Park, Minsu Jang, Jaeyeon Lee, and Jaehong Kim. “ETRI-Activity3D: A Large-Scale RGB-D Dataset for Robots to Recognize Daily Activities of the Elderly”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2020, Las Vegas, NV, USA, October 24, 2020 - January 24, 2021*. IEEE, 2020, pp. 10990–10997. DOI: 10.1109/IROS45743.2020.9341160.
- [JM19] Bhavan Jasani and Afshaan Mazagonwalla. “Skeleton based Zero Shot Action Recognition in Joint Pose-Language Semantic Space”. In: *CoRR abs/1911.11344* (2019). arXiv: 1911.11344.
- [Joh73] Gunnar Johansson. “Visual perception of biological motion and a model for its analysis”. In: *Perception & psychophysics* 14.2 (1973), pp. 201–211.
- [Joz+20] Hamid Reza Vaezi Joz, Amirreza Shaban, Michael L. Iuzzolino, and Kazuhito Koishida. “MMTM: Multimodal Transfer Module for CNN Fusion”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. IEEE, 2020, pp. 13286–13296. DOI: 10.1109/CVPR42600.2020.01330.
- [Kad+20] Abhishek Kadian, Joanne Truong, Aaron Gokaslan, Alexander Clegg, Erik Wijmans, Stefan Lee, Manolis Savva, Sonia Chernova, and Dhruv Batra. “Sim2Real Predictivity: Does Evaluation in Simulation Predict Real-World Performance?” In: *IEEE Robotics Autom. Lett.* 5.4 (2020), pp. 6670–6677. DOI: 10.1109/LRA.2020.3013848.
- [KAS14] Hilde Kuehne, Ali Bilgin Arslan, and Thomas Serre. “The Language of Actions: Recovering the Syntax and Semantics of Goal-Directed Human Activities”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 2014, pp. 780–787. DOI: 10.1109/CVPR.2014.105.
- [Kay+17] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. “The Kinetics Human Action Video Dataset”. In: *CoRR abs/1705.06950* (2017). arXiv: 1705.06950.
- [KB15] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015.

- [KBA19] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. “PifPaf: Composite Fields for Human Pose Estimation”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 11977–11986. DOI: 10.1109/CVPR.2019.01225.
- [Ke+17] Qihong Ke, Mohammed Bennamoun, Senjian An, Ferdous Ahmed Sohel, and Farid Boussaïd. “A New Representation of Skeleton Sequences for 3D Action Recognition”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 4570–4579. DOI: 10.1109/CVPR.2017.486.
- [Ke+18] Qihong Ke, Mohammed Bennamoun, Senjian An, Ferdous Ahmed Sohel, and Farid Boussaïd. “Learning Clip Representations for Skeleton-Based 3D Action Recognition”. In: *IEEE Trans. Image Process.* 27.6 (2018), pp. 2842–2855. DOI: 10.1109/TIP.2018.2812099.
- [Keb+20] Parham M. Kebria, Abbas Khosravi, Syed Moshfeq Salaken, and Saeid Nahavandi. “Deep imitation learning for autonomous vehicles based on convolutional neural networks”. In: *IEEE CAA J. Autom. Sinica* 7.1 (2020), pp. 82–95. DOI: 10.1109/jas.2019.1911825.
- [KF18] Yu Kong and Yun Fu. “Human Action Recognition and Prediction: A Survey”. In: *CoRR* abs/1806.11230 (2018). arXiv: 1806.11230.
- [KGS13] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. “Learning human activities and object affordances from RGB-D videos”. In: *I. J. Robotics Res.* 32.8 (2013), pp. 951–970. DOI: 10.1177/0278364913478446.
- [KGS16] Hilde Kuehne, Juergen Gall, and Thomas Serre. “An end-to-end generative framework for video segmentation and recognition”. In: *2016 IEEE Winter Conference on Applications of Computer Vision, WACV 2016, Lake Placid, NY, USA, March 7-10, 2016*. IEEE Computer Society, 2016, pp. 1–8. DOI: 10.1109/WACV.2016.7477701.
- [KH04] Nathan P. Koenig and Andrew Howard. “Design and use paradigms for Gazebo, an open-source multi-robot simulator”. In: *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, September 28 - October 2, 2004*. IEEE, 2004, pp. 2149–2154. DOI: 10.1109/IROS.2004.1389727.

- [KHW11] Orit Kliper-Gross, Tal Hassner, and Lior Wolf. “One Shot Similarity Metric Learning for Action Recognition”. In: *Similarity-Based Pattern Recognition - First International Workshop, SIMBAD 2011, Venice, Italy, September 28-30, 2011. Proceedings*. Ed. by Marcello Pelillo and Edwin R. Hancock. Vol. 7005. Lecture Notes in Computer Science. Springer, 2011, pp. 31–45. DOI: 10.1007/978-3-642-24471-1_3.
- [KO11] Bekir Karlik and A Vehbi Olgac. “Performance analysis of various activation functions in generalized MLP architectures of neural networks”. In: *International Journal of Artificial Intelligence and Expert Systems* 1.4 (2011), pp. 111–122.
- [Kon+19] Quan Kong, Ziming Wu, Ziwei Deng, Martin Klinkigt, Bin Tong, and Tomokazu Murakami. “MMAct: A Large-Scale Dataset for Cross Modal Human Action Understanding”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 8657–8666. DOI: 10.1109/ICCV.2019.00875.
- [Kot+19] Shashank Kotyan, Nishant Kumar, Pankaj Kumar Sahu, and Venkanna Udutalapally. “HAUAR: Home Automation Using Action Recognition”. In: *CoRR* abs/1904.10354 (2019). arXiv: 1904.10354.
- [KR17] Tae Soo Kim and Austin Reiter. “Interpretable 3D Human Action Analysis with Temporal Convolutional Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1623–1631. DOI: 10.1109/CVPRW.2017.207.
- [Krü+07] Volker Krüger, Danica Kragic, Aleš Ude, and Christopher Geib. “The meaning of action: A review on action recognition and mapping”. In: *Advanced Robotics* 21.13 (2007), pp. 1473–1501. DOI: 10.1163-156855307782148578.
- [KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*. Ed. by Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger. 2012, pp. 1106–1114.
- [Küm+09] Rainer Kümmerle, Bastian Steder, Christian Dornhege, Michael Ruhnke, Giorgio Grisetti, Cyrill Stachniss, and Alexander Kleiner. “On measuring the accuracy of SLAM algorithms”. In: *Auton. Robots* 27.4 (2009), pp. 387–407. DOI: 10.1007/s10514-009-9155-6.

- [KW17] Thomas N. Kipf and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks”. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.
- [LAM19] Jian Liu, Naveed Akhtar, and Ajmal Mian. “Skepxels: Spatio-temporal Image Representation of Human Skeleton Joints for Action Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019.
- [Las+17] Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. “DART: Noise Injection for Robust Imitation Learning”. In: *Proceedings of Machine Learning Research 78 (Nov. 2017)*. Ed. by Sergey Levine, Vincent Vanhoucke, and Ken Goldberg, pp. 143–156.
- [Lea+17] Colin Lea, Michael D. Flynn, René Vidal, Austin Reiter, and Gregory D. Hager. “Temporal Convolutional Networks for Action Segmentation and Detection”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1003–1012. DOI: 10.1109/CVPR.2017.113.
- [LeC+89] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. “Back-propagation Applied to Handwritten Zip Code Recognition”. In: *Neural Comput.* 1.4 (1989), pp. 541–551. DOI: 10.1162/neco.1989.1.4.541.
- [Lei+17] Jürgen Leitner, Adam W. Tow, Niko Sünderhauf, Jake E. Dean, Joseph W. Durham, Matthew Cooper, Markus Eich, Christopher F. Lehnert, Ruben Mangels, Christopher McCool, Peter T. Kujala, Lachlan Nicholson, Trung Pham, James Sergeant, Liao Wu, Fangyi Zhang, Ben Upcroft, and Peter I. Corke. “The ACRV picking benchmark: A robotic shelf picking benchmark to foster reproducible research”. In: *2017 IEEE International Conference on Robotics and Automation, ICRA 2017, Singapore, Singapore, May 29 - June 3, 2017*. IEEE, 2017, pp. 4705–4712. DOI: 10.1109/ICRA.2017.7989545.
- [LH17] Ilya Loshchilov and Frank Hutter. “SGDR: Stochastic Gradient Descent with Warm Restarts”. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

- [Li+16] Yanghao Li, Cuiling Lan, Junliang Xing, Wenjun Zeng, Chunfeng Yuan, and Jiaying Liu. “Online Human Action Detection Using Joint Classification-Regression Recurrent Neural Networks”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VII*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling. Vol. 9911. Lecture Notes in Computer Science. Springer, 2016, pp. 203–220. DOI: 10.1007/978-3-319-46478-7_13.
- [Li+17] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. “Skeleton-based action recognition with convolutional neural networks”. In: *2017 IEEE International Conference on Multimedia & Expo Workshops, ICME Workshops, Hong Kong, China, July 10-14, 2017*. IEEE Computer Society, 2017, pp. 597–600. DOI: 10.1109/ICMEW.2017.8026285.
- [Li+18] Chao Li, Qiaoyong Zhong, Di Xie, and Shiliang Pu. “Co-occurrence Feature Learning from Skeleton Data for Action Recognition and Detection with Hierarchical Aggregation”. In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. Ed. by Jérôme Lang. ijcai.org, 2018, pp. 786–792. DOI: 10.24963/ijcai.2018/109.
- [Li+19a] Bin Li, Xi Li, Zhongfei Zhang, and Fei Wu. “Spatio-Temporal Graph Routing for Skeleton-Based Action Recognition”. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 8561–8568. DOI: 10.1609/aaai.v33i01.33018561.
- [Li+19b] Tianhong Li, Lijie Fan, Mingmin Zhao, Yingcheng Liu, and Dina Katabi. “Making the Invisible Visible: Action Recognition Through Walls and Occlusions”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 872–881. DOI: 10.1109/ICCV.2019.00096.
- [Li+20a] Jianan Li, Xuemei Xie, Qingzhe Pan, Yuhan Cao, Zhifu Zhao, and Guangming Shi. “SGM-Net: Skeleton-guided multimodal network for action recognition”. In: *Pattern Recognit.* 104 (2020), p. 107356. DOI: 10.1016/j.patcog.2020.107356.

- [Li+20b] Tianjiao Li, Jun Liu, Wei Zhang, and Lingyu Duan. “HARD-Net: Hardness-AwaRe Discrimination Network for 3D Early Activity Prediction”. In: *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XI*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12356. Lecture Notes in Computer Science. Springer, 2020, pp. 420–436. DOI: 10.1007/978-3-030-58621-8_25.
- [Li+21] Tianjiao Li, Jun Liu, Wei Zhang, Yun Ni, Wenqian Wang, and Zhiheng Li. “UAV-Human: A Large Benchmark for Human Behavior Understanding With Unmanned Aerial Vehicles”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 16266–16275.
- [Lim+16] Pedro U. Lima, Daniele Nardi, Gerhard K. Kraetzschmar, Rainer Bischoff, and Matteo Matteucci. “RoCKIn and the European Robotics League: Building on RoboCup Best Practices to Promote Robot Competitions in Europe”. In: *RoboCup 2016: Robot World Cup XX [Leipzig, Germany, June 30 - July 4, 2016]*. Ed. by Sven Behnke, Raymond Sheh, Sanem Sariel, and Daniel D. Lee. Vol. 9776. Lecture Notes in Computer Science. Springer, 2016, pp. 181–192. DOI: 10.1007/978-3-319-68792-6_15.
- [Lin+14] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. “Microsoft COCO: Common Objects in Context”. In: *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*. 2014, pp. 740–755. DOI: 10.1007/978-3-319-10602-1_48.
- [Lin+19] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. “BMN: Boundary-Matching Network for Temporal Action Proposal Generation”. In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 3888–3897. DOI: 10.1109/ICCV.2019.00399.
- [Liu+16a] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. “Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling. Vol. 9907. Lecture Notes in Computer Science. Springer, 2016, pp. 816–833. DOI: 10.1007/978-3-319-46487-9_50.

- [Liu+16b] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. “SSD: Single Shot MultiBox Detector”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling. Vol. 9905. Lecture Notes in Computer Science. Springer, 2016, pp. 21–37. DOI: 10.1007/978-3-319-46448-0_2.
- [Liu+17a] Chunhui Liu, Yueyu Hu, Yanghao Li, Sijie Song, and Jiaying Liu. “PKU-MMD: A Large Scale Benchmark for Skeleton-Based Human Action Understanding”. In: *Proceedings of the Workshop on Visual Analysis in Smart and Connected Communities, VSCC@MM 2017, Mountain View, CA, USA, October 23, 2017*. Ed. by Xiaobai Liu, Yadong Mu, Yu-Gang Jiang, and Jiebo Luo. ACM, 2017, pp. 1–8. DOI: 10.1145/3132734.3132739.
- [Liu+17b] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C. Kot. “Global Context-Aware Attention LSTM Networks for 3D Action Recognition”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 3671–3680. DOI: 10.1109/CVPR.2017.391.
- [Liu+18a] Jun Liu, Amir Shahroudy, Dong Xu, Alex C. Kot, and Gang Wang. “Skeleton-Based Action Recognition Using Spatio-Temporal LSTM Network with Trust Gates”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 40.12 (2018), pp. 3007–3021. DOI: 10.1109/TPAMI.2017.2771306.
- [Liu+18b] Jun Liu, Gang Wang, Ling-Yu Duan, Kamila Abdiyeva, and Alex C. Kot. “Skeleton-Based Human Action Recognition With Global Context-Aware Attention LSTM Networks”. In: *IEEE Trans. Image Process.* 27.4 (2018), pp. 1586–1599. DOI: 10.1109/TIP.2017.2785279.
- [Liu+20a] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. “NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 42.10 (2020), pp. 2684–2701. DOI: 10.1109/TPAMI.2019.2916873.
- [Liu+20b] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. “Disentangling and Unifying Graph Convolutions for Skeleton-Based Action Recognition”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 143–152.

- [Liu+21a] Di Liu, Hui Xu, Jianzhong Wang, Yinghua Lu, Jun Kong, and Miao Qi. “Adaptive Attention Memory Graph Convolutional Networks for Skeleton-Based Action Recognition”. In: *Sensors* 21.20 (2021), p. 6761. DOI: 10.3390/s21206761.
- [Liu+21b] Yang Liu, Keze Wang, Guanbin Li, and Liang Lin. “Semantics-Aware Adaptive Knowledge Distillation for Sensor-to-Vision Action Recognition”. In: *IEEE Trans. Image Process.* 30 (2021), pp. 5573–5588. DOI: 10.1109/TIP.2021.3086590.
- [LKJ19] Tianshan Liu, Jun Kong, and Min Jiang. “RGB-D Action Recognition Using Multimodal Correlative Representation Learning Model”. In: *IEEE Sensors Journal* 19.5 (Mar. 2019), pp. 1862–1872. DOI: 10.1109/JSEN.2018.2884443.
- [LKL14] Martin Längkvist, Lars Karlsson, and Amy Loutfi. “A review of unsupervised feature learning and deep learning for time-series modeling”. In: *Pattern Recognit. Lett.* 42 (2014), pp. 11–24. DOI: 10.1016/j.patrec.2014.01.008.
- [LKL93] Ren C. Luo, Michael G. Kay, and W. Gary Lee. “Multisensor integration and fusion: Issues, approaches, and future trends”. In: *Robotics, Mechatronics and Manufacturing Systems*. Ed. by Toshi Takamori and Kazuo Tsuchiya. Elsevier, 1993, pp. 161–169. DOI: 10.1016/b978-0-444-89700-8.50030-0.
- [LLC17] Mengyuan Liu, Hong Liu, and Chen Chen. “Enhanced skeleton visualization for view invariant human action recognition”. In: *Pattern Recognit.* 68 (2017), pp. 346–362. DOI: 10.1016/j.patcog.2017.02.030.
- [LRJ19] Paula Lago, Claudia Roncancio, and Claudia Jiménez-Guarín. “Learning and managing context enriched behavior patterns in smart homes”. In: *Future Gener. Comput. Syst.* 91 (2019), pp. 191–205. DOI: 10.1016/j.future.2018.09.004.
- [Lug+19] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. “MediaPipe: A Framework for Building Perception - Pipelines”. In: *CoRR* abs/1906.08172 (2019). arXiv: 1906.08172.
- [Luo+18] Zelun Luo, Jun-Ting Hsieh, Lu Jiang, Juan Carlos Niebles, and Li Fei-Fei. “Graph Distillation for Action Detection with Privileged Modalities”. In: *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIV*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss.

- Vol. 11218. Lecture Notes in Computer Science. Springer, 2018, pp. 174–192. DOI: 10.1007/978-3-030-01264-9_11.
- [LY18] Mengyuan Liu and Junsong Yuan. “Recognizing Human Actions as the Evolution of Pose Estimation Maps”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018, pp. 1159–1168. DOI: 10.1109/CVPR.2018.00127.
- [Man+18] Kevis-Kokitsi Maninis, Sergi Caelles, Jordi Pont-Tuset, and Luc Van Gool. “Deep Extreme Cut: From Extreme Points to Object Segmentation”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 616–625. DOI: 10.1109/CVPR.2018.00071.
- [Mar+20] Fabio Martinelli, Francesco Mercaldo, Albina Orlando, Vittoria Nardone, Antonella Santone, and Arun Kumar Sangaiah. “Human behavior characterization for driving style recognition in vehicle system”. In: *Comput. Electr. Eng.* 83 (2020), p. 102504. DOI: 10.1016/j.compeleceng.2017.12.050.
- [Mar63] Donald W Marquardt. “An algorithm for least-squares estimation of non-linear parameters”. In: *Journal of the society for Industrial and Applied Mathematics* 11.2 (1963), pp. 431–441.
- [MBL20a] Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. “A Metric Learning Reality Check”. In: *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXV*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12370. Lecture Notes in Computer Science. Springer, 2020, pp. 681–699. DOI: 10.1007/978-3-030-58595-2_41.
- [MBL20b] Kevin Musgrave, Serge J. Belongie, and Ser-Nam Lim. “PyTorch Metric Learning”. In: *CoRR abs/2008.09164* (2020). arXiv: 2008.09164.
- [McI+18] Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. “UMAP: Uniform Manifold Approximation and Projection”. In: *J. Open Source Softw.* 3.29 (2018), p. 861. DOI: 10.21105/joss.00861.
- [Meh+20] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Pascal Fua, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt. “XNect: real-time multi-person 3D motion capture with a single RGB camera”. In: *ACM Trans. Graph.* 39.4 (2020), p. 82. DOI: 10.1145/3386569.3392410.

- [MI17] Yoshiaki Mizuchi and Tetsunari Inamura. “Cloud-based multimodal human-robot interaction simulator utilizing ROS and unity frameworks”. In: *IEEE/SICE International Symposium on System Integration, SII 2017, Taipei, Taiwan, December 11-14, 2017*. IEEE, 2017, pp. 948–955. DOI: 10.1109/SII.2017.8279345.
- [Mil+16] Anton Milan, Laura Leal-Taixé, Ian D. Reid, Stefan Roth, and Konrad Schindler. “MOT16: A Benchmark for Multi-Object Tracking”. In: *CoRR* abs/1603.00831 (2016). arXiv: 1603.00831.
- [Mis+18] Ashish Mishra, Vinay Kumar Verma, M. Shiva Krishna Reddy, Arul Kumar Subramaniam, Piyush Rai, and Anurag Mittal. “A Generative Approach to Zero-Shot and Few-Shot Action Recognition”. In: *2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018*. IEEE Computer Society, 2018, pp. 372–380. DOI: 10.1109/WACV.2018.00047.
- [MN78] David Marr and Herbert Keith Nishihara. “Representation and recognition of the spatial organization of three-dimensional shapes”. In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 200.1140 (1978), pp. 269–294.
- [MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008. ISBN: 978-0-521-86571-5. DOI: 10.1017/CBO9780511809071.
- [MSM17] Dmytro Mishkin, Nikolay Sergievskiy, and Jiri Matas. “Systematic evaluation of convolution neural network advances on the Imagenet”. In: *Comput. Vis. Image Underst.* 161 (2017), pp. 11–19. DOI: 10.1016/j.cviu.2017.05.007.
- [MT16] Behrooz Mahasseni and Sinisa Todorovic. “Regularizing Long Short - Term Memory with 3D Human-Skeleton Sequences for Action Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 3054–3062. DOI: 10.1109/CVPR.2016.333.
- [Mur12] Kevin P. Murphy. *Machine learning - a probabilistic perspective*. Adaptive computation and machine learning series. MIT Press, 2012. ISBN: 0262018020. DOI: 10.5555/2380985.
- [MUS16] Éric Marchand, Hideaki Uchiyama, and Fabien Spindler. “Pose Estimation for Augmented Reality: A Hands-On Survey”. In: *IEEE Trans. Vis. Comput. Graph.* 22.12 (2016), pp. 2633–2651. DOI: 10.1109/TVCG.2015.2513408.

- [NGC15] Qin Ni, Ana-Belén García-Hernando, and Iván Pau de la Cruz. “The Elderly’s Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development”. In: *Sensors* 15.5 (2015), pp. 11312–11362. DOI: 10.3390/s150511312.
- [NH10] Vinod Nair and Geoffrey E. Hinton. “Rectified Linear Units Improve Restricted Boltzmann Machines”. In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*. Ed. by Johannes Fürnkranz and Thorsten Joachims. Omnipress, 2010, pp. 807–814.
- [Niu+04] Wei Niu, Jiao Long, Dan Han, and Yuan-Fang Wang. “Human activity detection and recognition for video surveillance”. In: *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo, ICME 2004, 27-30 June 2004, Taipei, Taiwan*. IEEE Computer Society, 2004, pp. 719–722.
- [Nou+07] Norbert Noury, Anthony Fleury, Pierre Rumeau, AK Bourke, GO Laighin, Vincent Rialle, and JE Lundy. “Fall detection-principles and methods”. In: *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. 2007, pp. 1663–1666.
- [NVP18] Will Norcliffe-Brown, Stathis Vafeias, and Sarah Parisot. “Learning Conditioned Graph Structures for Interpretable Visual Question Answering”. In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett. 2018, pp. 8344–8353.
- [NYK09] Bingbing Ni, Shuicheng Yan, and Ashraf A. Kassim. “Recognizing human group activities with localized causalities”. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*. IEEE Computer Society, 2009, pp. 1470–1477. DOI: 10.1109/CVPR.2009.5206853.
- [Ope+19] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei Zhang. “Solving Rubik’s Cube with a Robot Hand”. In: *CoRR abs/1910.07113* (2019). arXiv: 1910.07113.
- [Opp+97] Alan V Oppenheim, Alan S Willsky, Syed Hamid Nawab, Gloria Mata Hernández, et al. *Signals & systems*. Pearson Educación, 1997.

- [Osa+18] Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J. Andrew Bagnell, Pieter Abbeel, and Jan Peters. “An Algorithmic Perspective on Imitation Learning”. In: *Found. Trends Robotics* 7.1-2 (2018), pp. 1–179. DOI: 10.1561/23000000053.
- [OWW18] Jeeheh Oh, Jiakuan Wang, and Jenna Wiens. “Learning to Exploit Invariances in Clinical Time-Series Data using Sequence Transformer Networks”. In: *Proceedings of the Machine Learning for Healthcare Conference, MLHC 2018, 17-18 August 2018, Palo Alto, California*. Ed. by Finale Doshi-Velez, Jim Fackler, Ken Jung, David C. Kale, Rajesh Ranganath, Byron C. Wallace, and Jenna Wiens. Vol. 85. Proceedings of Machine Learning Research. PMLR, 2018, pp. 332–347.
- [Pai+18] Tom Le Paine, Sergio Gomez Colmenarejo, Ziyu Wang, Scott E. Reed, Yusuf Aytar, Tobias Pfaff, Matthew W. Hoffman, Gabriel Barth-Maron, Serkan Cabi, David Budden, and Nando de Freitas. “One-Shot High-Fidelity Imitation: Training Large-Scale Deep Nets with RL”. In: *CoRR* abs/1810.05017 (2018). arXiv: 1810.05017.
- [Pan+20] Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos A. Theodorou, and Byron Boots. “Imitation learning for agile autonomous driving”. In: *Int. J. Robotics Res.* 39.2-3 (2020). DOI: 10.1177/0278364919880273.
- [Pap+20] Konstantinos Papadopoulos, Enjie Ghorbel, Djamila Aouada, and Björn E. Ottersten. “Vertex Feature Encoding and Hierarchical Temporal Modeling in a Spatio-Temporal Graph Convolutional Network for Action Recognition”. In: *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*. IEEE, 2020, pp. 452–458. DOI: 10.1109/ICPR48806.2021.9413189.
- [Pas+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett. 2019, pp. 8024–8035.

- [PCD15] Pedro H. O. Pinheiro, Ronan Collobert, and Piotr Dollár. “Learning to Segment Object Candidates”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 1990–1998.
- [Pen+20] Wei Peng, Xiaopeng Hong, Haoyu Chen, and Guoying Zhao. “Learning Graph Convolutional Network for Skeleton-Based Human Action Recognition by Neural Searching”. In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 2020, pp. 2669–2676.
- [Pen+21] Kunyu Peng, Alina Roitberg, David Schneider, Marios Koulakis, Kailun Yang, and Rainer Stiefelwagen. “Affect-DML: Context-Aware One-Shot Recognition of Human Affect using Deep Metric Learning”. In: *16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021, Jodhpur, India, December 15-18, 2021*. IEEE, 2021, pp. 1–8. DOI: 10.1109/FG52635.2021.9666940.
- [Per+19] Juan-Manuel Perez-Rua, Valentin Vielzeuf, Stéphane Pateux, Moez Baccouche, and Frédéric Jurie. “MFAS: Multimodal Fusion Architecture Search”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 6966–6975. DOI: 10.1109/CVPR.2019.00713.
- [PMC15] Dimitri Palaz, Mathew Magimai-Doss, and Ronan Collobert. “Analysis of CNN-based speech recognition system using raw speech as input”. In: *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*. ISCA, 2015, pp. 11–15.
- [PN94] Ramprasad Polana and Randal Nelson. “Low level recognition of human motion (or how to get your man without finding his body parts)”. In: *Proceedings of 1994 IEEE Workshop on Motion of Non-rigid and Articulated Objects*. IEEE. 1994, pp. 77–82.
- [PR12] Hamed Pirsiavash and Deva Ramanan. “Detecting activities of daily living in first-person camera views”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*. 2012, pp. 2847–2854. DOI: 10.1109/CVPR.2012.6248010.

- [Rah+16] Hossein Rahmani, Arif Mahmood, Du Q. Huynh, and Ajmal S. Mian. “Histogram of Oriented Principal Components for Cross-View Action Recognition”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 38.12 (2016), pp. 2430–2443. DOI: 10.1109/TPAMI.2016.2533389.
- [Ram+14] Karinne Ramirez-Amaro, Tetsunari Inamura, Emmanuel C. Dean-Leon, Michael Beetz, and Gordon Cheng. “Bootstrapping humanoid robot skills by extracting semantic representations of human-like activities from virtual reality”. In: *14th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2014, Madrid, Spain, November 18-20, 2014*. IEEE, 2014, pp. 438–443. DOI: 10.1109/HUMANOIDS.2014.7041398.
- [Ran+20] Muhammad Asif Rana, Daphne Chen, Jacob Williams, Vivian Chu, Seyed Reza Ahmadzadeh, and Sonia Chernova. “Benchmark for Skill Learning from Demonstration: Impact of User Experience, Task Complexity, and Start Configuration on Performance”. In: *2020 IEEE International Conference on Robotics and Automation, ICRA 2020, Paris, France, May 31 - August 31, 2020*. IEEE, 2020, pp. 7561–7567. DOI: 10.1109/ICRA40945.2020.9197470.
- [Red+16] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. “You Only Look Once: Unified, Real-Time Object Detection”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 779–788. DOI: 10.1109/CVPR.2016.91.
- [Rez+19] Hamid Reza Tofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian D. Reid, and Silvio Savarese. “Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 658–666. DOI: 10.1109/CVPR.2019.00075.
- [RG16] Alexander Richard and Juergen Gall. “Temporal Action Detection Using a Statistical Language Model”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 3131–3140. DOI: 10.1109/CVPR.2016.341.
- [RGB11] Stéphane Ross, Geoffrey J. Gordon, and Drew Bagnell. “A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning”. In: *JMLR Proceedings 15 (2011)*. Ed. by Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudík, pp. 627–635.

- [Rig+05] Marco Rigolli, Quentin Williams, Mark J Gooding, and Michael Brady. “Driver behavioural classification from trajectory data”. In: *Proceedings. 2005 IEEE Intelligent Transportation Systems, 2005*. IEEE, 2005, pp. 889–894.
- [RMM18] Debaditya Roy, C. Krishna Mohan, and K. Sri Rama Murty. “Action Recognition Based on Discriminative Embedding of Actions Using Siamese Networks”. In: *2018 IEEE International Conference on Image Processing, ICIP 2018, Athens, Greece, October 7-10, 2018*. IEEE, 2018, pp. 3473–3477. DOI: 10.1109/ICIP.2018.8451226.
- [Rod+17] Mario Rodríguez, Carlos Orrite, Carlos Medrano, and Dimitrios Makris. “Fast Simplex-HMM for One-Shot Learning Activity Recognition”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2017, Honolulu, HI, USA, July 21-26, 2017*. IEEE Computer Society, 2017, pp. 1259–1266. DOI: 10.1109/CVPRW.2017.166.
- [Rot+20] Karsten Roth, Timo Milbich, Samarth Sinha, Prateek Gupta, Björn Ommer, and Joseph Paul Cohen. “Revisiting Training Strategies and Generalization Performance in Deep Metric Learning”. In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, 2020, pp. 8242–8252.
- [RY16] Mohsen Ramezani and Farzin Yaghmaee. “A review on human action analysis in videos for retrieval applications”. In: *Artif. Intell. Rev.* 46.4 (2016), pp. 485–514. DOI: 10.1007/s10462-016-9473-y.
- [Ryo+15] M. S. Ryoo, Thomas J. Fuchs, Lu Xia, Jake K. Aggarwal, and Larry H. Matthies. “Robot-Centric Activity Prediction from First-Person Videos: What Will They Do to Me?”. In: *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction, HRI 2015, Portland, OR, USA, March 2-5, 2015*. Ed. by Julie A. Adams, William D. Smart, Bilge Mutlu, and Leila Takayama. ACM, 2015, pp. 295–302. DOI: 10.1145/2696454.2696462.
- [Ryo+20] Michael S. Ryoo, A. J. Piergiovanni, Juhana Kangaspunta, and Anelia Angelova. “AssembleNet++: Assembling Modality Representations via Attention Connections”. In: *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XX*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12365. Lecture Notes in Computer Science. Springer, 2020, pp. 654–671. DOI: 10.1007/978-3-030-58565-5_39.

- [RZL18] Prajit Ramachandran, Barret Zoph, and Quoc V. Le. “Searching for Activation Functions”. In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net, 2018.
- [Sab+21] Alberto Sabater, Laura Santos, José Santos-Victor, Alexandre Bernardino, Luis Montesano, and Ana C. Murillo. “One-shot action recognition towards novel assistive therapies”. In: *CoRR abs/2102.08997 (2021)*. arXiv: 2102.08997.
- [Sad+18] Fereshteh Sadeghi, Alexander Toshev, Eric Jang, and Sergey Levine. “Sim2Real Viewpoint Invariant Visual Servoing by Recurrent Control”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 4691–4699. DOI: 10.1109/CVPR.2018.00493.
- [San+18] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. “MobileNetV2: Inverted Residuals and Linear Bottlenecks”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.
- [Sha+16] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. “NTU - RGB+D: A Large Scale Dataset for 3D Human Activity Analysis”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 1010–1019. DOI: 10.1109/CVPR.2016.115.
- [Shi+19a] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. “Skeleton-Based Action Recognition With Directed Graph Neural Networks”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 7912–7921. DOI: 10.1109/CVPR.2019.00810.
- [Shi+19b] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. “Two-Stream Adaptive Graph Convolutional Networks for Skeleton-Based Action Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 12026–12035. DOI: 10.1109/CVPR.2019.01230.

- [Sho+11] Jamie Shotton, Andrew W. Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. “Real-time human pose recognition in parts from single depth images”. In: *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*. IEEE Computer Society, 2011, pp. 1297–1304. DOI: 10.1109/CVPR.2011.5995316.
- [Shv+21] Nina Shvetsova, Brian Chen, Andrew Rouditchenko, Samuel Thomas, Brian Kingsbury, Rogério Feris, David Harwath, James R. Glass, and Hilde Kuehne. “Everything at Once - Multi-modal Fusion Transformer for Video Retrieval”. In: *CoRR abs/2112.04446 (2021)*. arXiv: 2112.04446.
- [Si+19] Chenyang Si, Wentao Chen, Wei Wang, Liang Wang, and Tieniu Tan. “An Attention Enhanced Graph Convolutional LSTM Network for Skeleton-Based Action Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 1227–1236. DOI: 10.1109/CVPR.2019.00132.
- [Sim+17] Tomas Simon, Hanbyul Joo, Iain A. Matthews, and Yaser Sheikh. “Hand Keypoint Detection in Single Images Using Multiview Bootstrapping”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*. 2017, pp. 4645–4653. DOI: 10.1109/CVPR.2017.494.
- [SKP15] Florian Schroff, Dmitry Kalenichenko, and James Philbin. “FaceNet: A unified embedding for face recognition and clustering”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 815–823. DOI: 10.1109/CVPR.2015.7298682.
- [SL17] Patrick Schäfer and Ulf Leser. “Fast and Accurate Time Series Classification with WEASEL”. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*. Ed. by Ee-Peng Lim, Marianne Winslett, Mark Sanderson, Ada Wai-Chee Fu, Jimeng Sun, J. Shane Culpepper, Eric Lo, Joyce C. Ho, Debora Donato, Rakesh Agrawal, Yu Zheng, Carlos Castillo, Aixin Sun, Vincent S. Tseng, and Chenliang Li. ACM, 2017, pp. 637–646. DOI: 10.1145/3132847.3132980.
- [SMB10] Dominik Scherer, Andreas C. Müller, and Sven Behnke. “Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition”. In: *Artificial Neural Networks - ICANN 2010 - 20th International*

- Conference, Thessaloniki, Greece, September 15-18, 2010, Proceedings, Part III*. Ed. by Konstantinos I. Diamantaras, Wlodek Duch, and Lazaros S. Iliadis. Vol. 6354. Lecture Notes in Computer Science. Springer, 2010, pp. 92–101. DOI: 10.1007/978-3-642-15825-4_10.
- [Son+18] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. “Skeleton-Indexed Deep Multi-Modal Feature Learning for High Performance Human Action Recognition”. In: *2018 IEEE International Conference on Multimedia and Expo, ICME 2018, San Diego, CA, USA, July 23-27, 2018*. IEEE Computer Society, 2018, pp. 1–6. DOI: 10.1109/ICME.2018.8486486.
- [Son+20a] Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. “Stronger, Faster and More Explainable: A Graph Convolutional Baseline for Skeleton-based Action Recognition”. In: *MM ’20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*. Ed. by Chang Wen Chen, Rita Cucchiara, Xian-Sheng Hua, Guo-Jun Qi, Elisa Ricci, Zhengyou Zhang, and Roger Zimmermann. ACM, 2020, pp. 1625–1633. DOI: 10.1145/3394171.3413802.
- [Son+20b] Ziyang Song, Ziyi Yin, Zejian Yuan, Chong Zhang, Wanchao Chi, Yonggen Ling, and Shenghao Zhang. “Attention-Oriented Action Recognition for Real-Time Human-Robot Interaction”. In: *25th International Conference on Pattern Recognition, ICPR 2020, Virtual Event / Milan, Italy, January 10-15, 2021*. IEEE, 2020, pp. 7087–7094. DOI: 10.1109/ICPR48806.2021.9412346.
- [Son+21] Yi-Fan Song, Zhang Zhang, Caifeng Shan, and Liang Wang. “Constructing Stronger and Faster Baselines for Skeleton-based Action Recognition”. In: *CoRR abs/2106.15125 (2021)*. arXiv: 2106.15125.
- [ST17] Markus D. Solbach and John K. Tsotsos. “Vision-Based Fallen Person Detection for the Elderly”. In: *2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 1433–1442. DOI: 10.1109/ICCVW.2017.170.
- [Stu+12] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. “A benchmark for the evaluation of RGB-D SLAM systems”. In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*. IEEE, 2012, pp. 573–580. DOI: 10.1109/IROS.2012.6385773.

- [SYS14] Zhixin Shu, Kiwon Yun, and Dimitris Samaras. “Action Detection with Improved Dense Trajectories and Sliding Window”. In: *Computer Vision - ECCV 2014 Workshops - Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part I*. Ed. by Lourdes Agapito, Michael M. Bronstein, and Carsten Rother. Vol. 8925. Lecture Notes in Computer Science. Springer, 2014, pp. 541–551. DOI: 10.1007/978-3-319-16178-5_38.
- [SZ14] Cícero Nogueira dos Santos and Bianca Zadrozny. “Learning Character-level Representations for Part-of-Speech Tagging”. In: *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*. Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, 2014, pp. 1818–1826.
- [Sze+15] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. “Going deeper with convolutions”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. IEEE Computer Society, 2015, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.
- [Tah+20] Omid Taheri, Nima Ghorbani, Michael J. Black, and Dimitrios Tzionas. “GRAB: A Dataset of Whole-Body Human Grasping of Objects”. In: *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part IV*. Ed. by Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm. Vol. 12349. Lecture Notes in Computer Science. Springer, 2020, pp. 581–600. DOI: 10.1007/978-3-030-58548-8_34.
- [TG19] Fida Mohammad Thoker and Juergen Gall. “Cross-modal knowledge distillation for action recognition”. In: *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 6–10.
- [TH21] Santosh Thoduka and Nico Hochgeschwender. “Benchmarking Robots by Inducing Failures in Competition Scenarios”. In: *Digital Human Modeling and Applications in Health, Safety, Ergonomics and Risk Management. AI, Product and Service - 12th International Conference, DHM 2021, Held as Part of the 23rd HCI International Conference, HCII 2021, Virtual Event, July 24-29, 2021, Proceedings, Part II*. Ed. by Vincent G. Duffy. Vol. 12778. Lecture Notes in Computer Science. Springer, 2021, pp. 263–276. DOI: 10.1007/978-3-030-77820-0_20.
- [TJA18] Rajesh Kumar Tripathi, Anand Singh Jalal, and Subhash Chand Agrawal. “Suspicious human activity recognition: a review”. In: *Artif. Intell. Rev.* 50.2 (2018), pp. 283–339. DOI: 10.1007/s10462-017-9545-7.

- [TL19] Mingxing Tan and Quoc V. Le. “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 6105–6114.
- [Toy+20] Sam Toyer, Rohin Shah, Andrew Critch, and Stuart Russell. “The MAGICAL Benchmark for Robust Imitation”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. 2020.
- [Tra+15] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. “Learning Spatiotemporal Features with 3D Convolutional Networks”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 2015, pp. 4489–4497. DOI: 10.1109/ICCV.2015.510.
- [Tra+18] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. “A Closer Look at Spatiotemporal Convolutions for Action Recognition”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. IEEE Computer Society, 2018, pp. 6450–6459. DOI: 10.1109/CVPR.2018.00675.
- [TS14] Alexander Toshev and Christian Szegedy. “DeepPose: Human Pose Estimation via Deep Neural Networks”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 2014, pp. 1653–1660. DOI: 10.1109/CVPR.2014.214.
- [Ull+19] Amin Ullah, Khan Muhammad, Ijaz Ul Haq, and Sung Wook Baik. “Action recognition using optimized deep autoencoder and CNN for surveillance data streams of non-stationary environments”. In: *Future Gener. Comput. Syst.* 96 (2019), pp. 386–397. DOI: 10.1016/j.future.2019.01.029.
- [VAC14] Raviteja Vemulapalli, Felipe Arrate, and Rama Chellappa. “Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 2014, pp. 588–595. DOI: 10.1109/CVPR.2014.82.

- [Vas+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett. 2017, pp. 5998–6008.
- [VCS18] Valeria Villani, Beatrice Capelli, and Lorenzo Sabattini. “Use of Virtual Reality for the Evaluation of Human-Robot Interaction Systems in Complex Scenarios”. In: *27th IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN 2018, Nanjing, China, August 27-31, 2018*. IEEE, 2018, pp. 422–427. DOI: 10.1109/ROMAN.2018.8525738.
- [Wad17] Kazuyoshi Wada. “New robot technology challenge for convenience store”. In: *IEEE/SICE International Symposium on System Integration, SII 2017, Taipei, Taiwan, December 11-14, 2017*. IEEE, 2017, pp. 1086–1091. DOI: 10.1109/SII.2017.8279367.
- [Wan+13] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. “Dense Trajectories and Motion Boundary Descriptors for Action Recognition”. In: *Int. J. Comput. Vis.* 103.1 (2013), pp. 60–79. DOI: 10.1007/s11263-012-0594-8.
- [Wan+14] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. “Learning Fine-Grained Image Similarity with Deep Ranking”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, 2014, pp. 1386–1393. DOI: 10.1109/CVPR.2014.180.
- [Wan+16] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. “Temporal Segment Networks: Towards Good Practices for Deep Action Recognition”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling. Vol. 9912. Lecture Notes in Computer Science. Springer, 2016, pp. 20–36. DOI: 10.1007/978-3-319-46484-8_2.
- [Wan+18a] Pichao Wang, Wanqing Li, Chuankun Li, and Yonghong Hou. “Action recognition based on joint trajectory maps with convolutional neural networks”. In: *Knowl. Based Syst.* 158 (2018), pp. 43–53. DOI: 10.1016/j.knosys.2018.05.029.

- [Wan+18b] Pichao Wang, Wanqing Li, Jun Wan, Philip Ogunbona, and Xinwang Liu. “Cooperative Training of Deep Aggregation Networks for RGB-D Action Recognition”. In: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. Ed. by Sheila A. McIlraith and Kilian Q. Weinberger. AAAI Press, 2018, pp. 7404–7411.
- [Wan+18c] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. “Non-Local Neural Networks”. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 7794–7803. DOI: 10.1109/CVPR.2018.00813.
- [Wan+19a] Fei Wang, Jianwei Feng, Yinliang Zhao, Xiaobin Zhang, Shiyuan Zhang, and Jinsong Han. “Joint Activity Recognition and Indoor Localization With WiFi Fingerprints”. In: *IEEE Access* 7 (2019), pp. 80058–80068. DOI: 10.1109/ACCESS.2019.2923743.
- [Wan+19b] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R. Scott. “Multi-Similarity Loss With General Pair Weighting for Deep Metric Learning”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 5022–5030. DOI: 10.1109/CVPR.2019.00516.
- [WB18] Nicolai Wojke and Alex Bewley. “Deep Cosine Metric Learning for Person Re-identification”. In: *2018 IEEE Winter Conference on Applications of Computer Vision, WACV 2018, Lake Tahoe, NV, USA, March 12-15, 2018*. IEEE Computer Society, 2018, pp. 748–756. DOI: 10.1109/WACV.2018.00087.
- [Wei+16] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. “Convolutional Pose Machines”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 4724–4732. DOI: 10.1109/CVPR.2016.511.
- [Wen+19] Yu-Hui Wen, Lin Gao, Hongbo Fu, Fang-Lue Zhang, and Shihong Xia. “Graph CNNs with Motif and Variable Temporal Block for Skeleton-Based Action Recognition”. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019,*

- Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 8989–8996. DOI: 10.1609/aaai.v33i01.33018989.
- [WG18] Xiaolong Wang and Abhinav Gupta. “Videos as Space-Time Region Graphs”. In: *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part V*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss. Vol. 11209. Lecture Notes in Computer Science. Springer, 2018, pp. 413–431. DOI: 10.1007/978-3-030-01228-1_25.
- [Wil+12] Arnold Wiliem, Vamsi Krishna Madasu, Wageeh W. Boles, and Prasad K. D. V. Yarlagadda. “A suspicious behaviour detection using a context space model for smart surveillance systems”. In: *Comput. Vis. Image Underst.* 116.2 (2012), pp. 194–209. DOI: 10.1016/j.cviu.2011.10.001.
- [Wis+09] Thomas Wisspeintner, Tijn Van Der Zant, Luca Iocchi, and Stefan Schiffer. “RoboCup@ Home: Scientific competition and benchmarking for domestic service robots”. In: *Interaction Studies* 10.3 (2009), pp. 392–426.
- [WLY13] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. “Online Object Tracking: A Benchmark”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*. IEEE Computer Society, 2013, pp. 2411–2418. DOI: 10.1109/CVPR.2013.312.
- [WMS16] Philippe Weinzaepfel, Xavier Martin, and Cordelia Schmid. “Human Action Localization with Sparse Spatial Supervision”. In: *arXiv preprint arXiv:1605.05197* (2016).
- [WS09] Kilian Q. Weinberger and Lawrence K. Saul. “Distance Metric Learning for Large Margin Nearest Neighbor Classification”. In: *J. Mach. Learn. Res.* 10 (2009), pp. 207–244. ISSN: 1532-4435. DOI: 10.5555/1577069.1577078.
- [WS13] Heng Wang and Cordelia Schmid. “Action Recognition with Improved Trajectories”. In: *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*. IEEE Computer Society, 2013, pp. 3551–3558. DOI: 10.1109/ICCV.2013.441.
- [WW18] Hongsong Wang and Liang Wang. “Beyond Joints: Learning Representations From Primitive Geometries for Skeleton-Based Action Recognition and Detection”. In: *IEEE Trans. Image Process.* 27.9 (2018), pp. 4382–4394. DOI: 10.1109/TIP.2018.2837386.

- [WYD17] Zhihua Wang, Zhaochu Yang, and Tao Dong. “A Review of Wearable Technologies for Elderly Care that Can Accurately Track Indoor Position, Recognize Physical Activities and Monitor Vital Signs in Real Time”. In: *Sensors* 17.2 (2017), p. 341. DOI: 10.3390/s17020341.
- [Xu+15] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. “Empirical Evaluation of Rectified Activations in Convolutional Network”. In: *CoRR* abs/1505.00853 (2015).
- [Xu+18] Yangyang Xu, Jun Cheng, Lei Wang, Haiying Xia, Feng Liu, and Dapeng Tao. “Ensemble One-Dimensional Convolution Neural Networks for Skeleton-Based Action Recognition”. In: *IEEE Signal Process. Lett.* 25.7 (2018), pp. 1044–1048. DOI: 10.1109/LSP.2018.2841649.
- [XWW18] Bin Xiao, Haiping Wu, and Yichen Wei. “Simple Baselines for Human Pose Estimation and Tracking”. In: *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VI*. Ed. by Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss. Vol. 11210. Lecture Notes in Computer Science. Springer, 2018, pp. 472–487. DOI: 10.1007/978-3-030-01231-1_29.
- [Yan+15] Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, and Shonali Krishnaswamy. “Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition”. In: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*. Ed. by Qiang Yang and Michael J. Wooldridge. AAAI Press, 2015, pp. 3995–4001.
- [Yan+19] Zhengyuan Yang, Yuncheng Li, Jianchao Yang, and Jiebo Luo. “Action Recognition With Spatio-Temporal Visual Attention on Skeleton Image Sequences”. In: *IEEE Trans. Circuits Syst. Video Technol.* 29.8 (2019), pp. 2405–2415. DOI: 10.1109/TCSVT.2018.2864148.
- [Yi+14] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z. Li. “Deep Metric Learning for Person Re-identification”. In: *22nd International Conference on Pattern Recognition, ICPR 2014, Stockholm, Sweden, August 24-28, 2014*. IEEE Computer Society, 2014, pp. 34–39. DOI: 10.1109/ICPR.2014.16.
- [Yok+19] Yasuyoshi Yokokohji, Yoshihiro Kawai, Mizuho Shibata, Yasumichi Aiyama, Shinya Kotosaka, Wataru Uemura, Akio Noda, Hiroki Dobashi, Takeshi Sakaguchi, and Kazuhito Yokoi. “Assembly Challenge: a robot competition of the Industrial Robotics Category, World Robot Summit - summary of the pre-competition in 2018”. In: *Adv. Robotics* 33.17 (2019), pp. 876–899. DOI: 10.1080/01691864.2019.1663609.

- [Yu+18] Tianhe Yu, Chelsea Finn, Sudeep Dasari, Annie Xie, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. “One-Shot Imitation from Observing Humans via Domain-Adaptive Meta-Learning”. In: *Robotics: Science and Systems XIV, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, June 26-30, 2018*. Ed. by Hadas Kress-Gazit, Siddhartha S. Srinivasa, Tom Howard, and Nikolay Atanasov. 2018. DOI: 10.15607/RSS.2018.XIV.002.
- [Yu+19] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. “Meta-World: A Benchmark and Evaluation for Multi-Task and Meta Reinforcement Learning”. In: *3rd Annual Conference on Robot Learning, CoRL 2019, Osaka, Japan, October 30 - November 1, 2019, Proceedings*. Ed. by Leslie Pack Kaelbling, Danica Kragic, and Komei Sugiura. Vol. 100. Proceedings of Machine Learning Research. PMLR, 2019, pp. 1094–1100.
- [YXL18] Sijie Yan, Yuanjun Xiong, and Dahua Lin. “Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition”. In: *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. Ed. by Sheila A. McIlraith and Kilian Q. Weinberger. AAAI Press, 2018, pp. 7444–7452.
- [ZBX18] Ying Zheng, Hong Bao, and Cheng Xu. “A method for improved pedestrian gesture recognition in self-driving cars”. In: *Australian Journal of Mechanical Engineering* 16.sup1 (2018), pp. 78–85. DOI: 10.1080/1448837X.2018.1545476. eprint: <https://doi.org/10.1080/1448837X.2018.1545476>.
- [ZC16] Jiakai Zhang and Kyunghyun Cho. “Query-Efficient Imitation Learning for End-to-End Autonomous Driving”. In: *CoRR* abs/1605.06450 (2016). arXiv: 1605.06450.
- [Zer+21] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. “A Transformer-based Framework for Multivariate Time Series Representation Learning”. In: *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. Ed. by Feida Zhu, Beng Chin Ooi, and Chunyan Miao. ACM, 2021, pp. 2114–2124. DOI: 10.1145/3447548.3467401.

- [Zha+16] Jing Zhang, Wanqing Li, Philip O. Ogunbona, Pichao Wang, and Chang Tang. “RGB-D-based action recognition datasets: A survey”. In: *Pattern Recognition* 60 (2016), pp. 86–105. DOI: 10.1016/j.patcog.2016.05.019.
- [Zha+17] Yifan Zhao, Lorenz Görne, Iek-Man Yuen, Dongpu Cao, Mark Sullman, Daniel J. Auger, Chen Lv, Huaji Wang, Rebecca Matthias, Lee Skrypchuk, and Alexandros Mouzakitis. “An Orientation Sensor-Based Head Tracking System for Driver Behaviour Monitoring”. In: *Sensors* 17.11 (2017), p. 2692. DOI: 10.3390/s17112692.
- [Zha+18] Tianhao Zhang, Zoe McCarthy, Owen Jow, Dennis Lee, Xi Chen, Ken Goldberg, and Pieter Abbeel. “Deep Imitation Learning for Complex Manipulation Tasks from Virtual Reality Teleoperation”. In: (2018), pp. 1–8. DOI: 10.1109/ICRA.2018.8461249.
- [Zha+19a] Hongbo Zhang, Yi-Xiang Zhang, Bineng Zhong, Qing Lei, Lijie Yang, Ji-Xiang Du, and Duan-Sheng Chen. “A Comprehensive Survey of Vision-Based Human Action Recognition Methods”. In: *Sensors* 19.5 (2019), p. 1005. DOI: 10.3390/s19051005.
- [Zha+19b] Rui Zhao, Wanru Xu, Hui Su, and Qiang Ji. “Bayesian Hierarchical Dynamic Model for Human Action Recognition”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, 7733–7742. DOI: 10.1109/CVPR.2019.00792.
- [Zha00] Zhengyou Zhang. “A Flexible New Technique for Camera Calibration”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 22.11 (2000), pp. 1330–1334. DOI: 10.1109/34.888718.
- [Zha12] Zhengyou Zhang. “Microsoft Kinect Sensor and Its Effect”. In: *IEEE Multimed.* 19.2 (2012), pp. 4–10. DOI: 10.1109/MMUL.2012.24.
- [Zhe+20] Ce Zheng, Wenhan Wu, Taojiannan Yang, Sijie Zhu, Chen Chen, Ruixu Liu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. “Deep Learning-Based Human Pose Estimation: A Survey”. In: *CoRR* abs/2012.13392 (2020). arXiv: 2012.13392.
- [ZL17] Barret Zoph and Quoc V. Le. “Neural Architecture Search with Reinforcement Learning”. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

- [Zol+17] Mohammadreza Zolfaghari, Gabriel L. Oliveira, Nima Sedaghat, and Thomas Brox. “Chained Multi-stream Networks Exploiting Pose, Motion, and Appearance for Action Classification and Detection”. In: *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2923–2932. DOI: 10.1109/ICCV.2017.316.
- [ZZL15] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. “Character-level Convolutional Networks for Text Classification”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett. 2015, pp. 649–657.