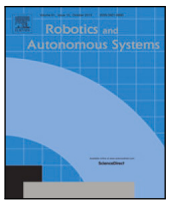
A general skeleton-based action and gesture recognition framework for human-robot collaboration

Robotics and Autonomous Systems 170 (2023) 104523

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



A general skeleton-based action and gesture recognition framework for human–robot collaboration

Matteo Terreran [a,*], Leonardo Barcellona [a,b], Stefano Ghidoni [a]

[a] Department of Information Engineering, University of Padova, Padova, Italy
[b] Politecnico di Torino, Torino, Italy

ARTICLE INFO

Keywords:
Human action recognition
Gesture recognition
3D pose estimation
Ensemble learning
Human–robot collaboration

ABSTRACT

Recognizing human actions is crucial for an effective and safe collaboration between humans and robots. For example, in a collaborative assembly task, human workers can use gestures to communicate with the robot, and the robot can use the recognized actions to anticipate the next steps in the assembly process, leading to improved safety and productivity. In this work, we propose a general framework for human action recognition based on 3D pose estimation and ensemble techniques, which allows to recognize both body actions and hand gestures. The framework relies on OpenPose and 2D to 3D lifting methods to estimate 3D joints for the human body and the hands, feeding then these joints into a set of graph convolutional networks based on the Shift-GCN architecture. The output scores of all networks are combined using an ensemble approach to predict the final human action. The proposed framework was evaluated on a custom dataset designed for human–robot collaboration tasks, named IAS-Lab Collaborative HAR dataset. The results showed that using an ensemble of action recognition models improves the accuracy and robustness of the overall system; moreover, the proposed framework can be easily specialized on different scenarios and achieve state-of-the-art results on the HRI30 dataset when coupled with an object detector or classifier.

1. Introduction

Human–robot collaboration (HRC) aims to a close and direct collaboration between robots and humans to reach higher productivity and ergonomics thanks to the synergy between human intelligence and robot mechanical power [1–3]. In such scenario, the robot must always be aware of the location and intentions of the human worker to prevent any potential dangerous situations and ensure the safety of the human partner. Additionally, by understanding the process step the human is working on, the robot can plan its actions properly, such as moving to a different area of the workspace or preparing parts and tools for the next stage of the assembly process.

Human action recognition (HAR) has been widely investigated in the literature to provide such awareness to the robot [4–6]. The actions typically considered are steps in an assembly sequence (e.g., picking up a part, placing a part, or screwing) or general actions like interacting, walking or standing still; all these actions involve various parts of the body and can be easily distinguished from each other. Some works instead, address human action recognition focusing on gestures, which are typically small movements of a few body parts, such as hands, used to convey information to the robot (e.g. move left, move right, stop). Given the difference in the parts involved, the problem of action recognition and gesture recognition are generally tackled separately with specialized methods and setups, such as using a camera that only frames the human hands in case of gesture recognition.

In this work, we address the problem of human action recognition in collaborative scenarios by proposing a general framework capable of recognizing both hands gestures and full-body actions (i.e., general movements involving the whole body). The framework was designed to be easily applied in various collaborative robotics scenarios, thus trying to meet two main requirements: (i) being general with respect to the set of actions to be recognized and (ii) being robust with respect to possible usage scenarios. For example, the same HAR system should recognize the actions of the human worker in different collaborative cells that may differ in viewpoint of the perception system or the tools employed by the worker during various stages of the assembly process.

Our framework relies on skeleton-based action recognition models, where 3D human pose (i.e., skeleton) is used as an intermediary representation between the action classifier and the raw image data. This allows to easily generalize on different scenarios and collaborative tasks, thanks to the robust representation of the 3D skeletons, which are independent on the viewpoints and unaffected by the scene context such as external objects, illumination and aesthetic differences of

* Corresponding author.
E-mail addresses: matteo.terreran@unipd.it (M. Terreran), leonardo.barcellona@phd.unipd.it (L. Barcellona), ghidoni@unipd.it (S. Ghidoni).

https://doi.org/10.1016/j.robot.2023.104523

Available online 16 September 2023

people such as clothes or skin color. Moreover, by using skeletons, the action classifier can focus on sequences of body poses that only describe human movements to learn a more general and robust representation of the actions of interest. Consider for example actions such as "picking up a piece to be assembled" or "grabbing a hammer from a toolbox", both actions share the same movements and could be considered a "pick" action despite the specific object being picked up, which could be classified separately using an object detector. Unlike other works in the literature, we did not restrict the action set to a specific application (e.g., take part A, place part B, take hammer from the toolbox), but we tried to generalize the most recurrent body actions and gestures in order to create a system capable of recognizing common actions in several collaborative tasks.

In a previous version of this work [7], the 3D human pose was estimated from 2D pose and RGB-D data by means of projection. Although this approach is commonly used in the literature [8–10] to obtain the 3D pose information of the person's body, in our previous work we pointed out that it is a very inaccurate solution to obtain the 3D pose of the hands, leading to many incomplete 3D skeletons due to missing hand joints. To alleviate such problem, our framework relies on an ensemble of different skeleton-based classifiers, each one trained to recognize actions on a subset of skeleton joints (e.g., body joints, hands joints); the use of dedicated classifiers allows to be robust to partial skeleton inputs with missing joints, and to handle together body actions and hand gestures. In this work, we further improve the robustness and generalization capabilities of the proposed HAR system by investigating different approaches to compute a more robust 3D pose information for the hands joint, such as "2D to 3D lifting" [11], and monocular 3D Human Shape estimation [12]. The result is an even more robust and flexible action recognition system, applicable even in scenarios where depth is not available but only RGB data.

The proposed system has been trained and evaluated on a dataset acquired on purpose in our laboratory, namely the IAS-LAB Collaborative HAR dataset,[1] which includes RGB-D videos of several subjects executing typical collaborative actions between human and robot. The system was further validated on the HRI30 dataset [13], a dataset of RGB videos acquired in a collaborative setting which includes many actions describing human movements coupled with tools and objects. The experiments demonstrated that our system is able to generalize to novel scenarios, even with no depth information and changes in the scene background and viewpoint; in addition, we proved that when coupled with a simple object classifier, the proposed HAR system is able to outperform state-of-the art methods on the HRI30 dataset.

Summarizing, the work presents the following main contributions: (i) a unified framework for human action and gesture recognition in a human–robot collaboration scenario; (ii) an experimental comparison of different ensembling techniques to improve the overall accuracy and robustness of the system; (iii) an experimental comparison of different 3D pose estimation methods to alleviate the missing joints problem for the hands; (iv) a novel RGB-D dateset for action recognition in a human–robot collaboration scenario, including both general actions and hand gestures, to further drive research in this field.

The remainder of the paper is organized as follows. Section 2 reviews the works related to action recognition, with a focus on human–robot collaboration scenarios. In Section 3 the main elements of our system are described in details. In Section 4 we present the action recognition dataset acquired in our laboratory, used to thoroughly evaluate the proposed system in Section 5. In Section 6 the system is further evaluated on the HRI30 dataset, proving its robustness on a different collaborative scenarios. Finally, in Section 7, conclusions are drawn and future directions of research identified.

[1] Available at http://robotics.dei.unipd.it/.

2. Related works

Human action recognition (HAR) is generally defined as the process of identifying and analyzing the movements of one or several parts of the human body, with many applications related to video surveillance, such as public events [14] and home monitoring [15,16], human–robot interaction [5,17], and safety monitoring in industry [18]. Human action recognition systems can be divided into two main categories: contact-based and vision-based methods. Contact-based methods involve physical interaction with sensors and devices such as accelerometers, multi-touch screens, body-mounted sensors, and wearable sensors [16,17,19]. Vision-based methods, on the other hand, use images or videos to recognize activities [20–22], and can utilize a single camera or a network of cameras to handle occlusions. Vision-based systems are considered non-intrusive as they do not require users to wear multiple devices, making them more suitable for real-world scenarios.

A significant challenge in vision-based action recognition is handling both the spatial and temporal dimensions, as actions are typically considered as a series of consecutive movements over time. Several methods have been proposed to address this challenge, including LSTMs [23,24], 3D-CNNs [25,26], and multi-stream 2D-CNNs [27,28]. 3D-CNNs use a sequence of RGB frames as input and employ 3D convolution kernels to analyze the temporal information. On the other hand, multi-stream CNNs have two branches in the network that analyze spatial and temporal information separately, using RGB frames and optical flow information as inputs, respectively. Recently, human body pose estimation models, like OpenPose [29], have achieved high performance, leading to an increasing number of researchers using 3D human body pose as input for graph convolutional networks (GCNs) [30–32]. Body pose is a more compact representation of both spatial and temporal information than images, resulting in GCN models like the Shift-GCN architecture [30] outperforming other methods on popular action recognition datasets [33,34].

It is particularly challenging to recognize actions when they are captured from different angles because there are so many variations in their representations. In [35] authors propose the use of dense optical flow as a local feature descriptor to maker their method robust under a wide range of viewpoint changes. Another commonly used approach to be robust to viewpoint change is the use of skeleton-based action recognition model [36,37], since they rely on compact data that are less affected by complex backgrounds and viewpoint changes representation.

2.1. Human 3D pose estimation

Human Pose Estimation (HPE) aims to estimate the position of human joints in an image or in the 3D space. Despite many recent works achieved impressive results for 2D HPE [29,38], 3D HPE is still an open challenge. The most common approach to obtain 3D pose information involves the use of RGB-D sensors, which provide RGB and depth frames. For each joint predicted by a 2D pose estimator (e.g., OpenPose [29]) on the RGB image, the corresponding 3D coordinates are computed by means of re-projection using the sensor intrinsic parameters and the depth information. However, this method is very sensitive to the quality of depth information, returning invalid or very inaccurate values in the case of hands or when only one side of the human body is visible.

Recently, researchers focused on monocular HPE, which estimates the 3D coordinates of human body joints from RGB images only. A notable solution is MeTRAbs [39], that uses volumetric heatmaps invariant to scale and truncation for directly estimating 3D poses without using prior knowledge on camera distance or anthropomorphic measures. A different approach to monocular HPE is the "2D to 3D lifting" [11,40–42], which computes the 3D pose by means of 2D pose only, without requiring any depth information. The core idea is that

2D HPE solutions are more robust to changes (e.g., light variation or pose variation), and they contain enough information to predict a good approximation of 3D poses. For example, VideoPose3D [41] makes use of sequences of 2D poses to resolve ambiguous human poses (i.e., poses sharing the same 2D projection). Li et al. [42] followed the same approach replacing temporal convolutions with vanilla transformers and proposing a transformer based module, called "strided transformer", to refine the predicted 3D poses. All the approaches just described do not explicitly consider any anthropometric parameters, which leads many researchers to tackle the 3D pose estimation problem by fitting a parametric model of the human body. SMPL [43] is a parametric model that maps shape, pose parameters, into mesh points of a person. SMPLify [12] fits SMPL models minimizing the distance between keypoints provided by a 2D pose estimator and the projected keypoints of the model. The same procedure is inherited by SMPLify-x [44], where authors improve the hands and the face expression incorporating respectively MANO [45] and FLAME [46] models.

2.2. Action recognition for human–robot collaboration

Human action recognition is widely used in various human–robot interaction settings such as social robotics and manufacturing industries. Typically, in social robotics the set of actions that needs to be recognized includes hand gestures and facial expressions to facilitate easy and efficient interaction with robots [4,47]. In [4], the authors proposed six gestures to enable communication between a human and a collaborative robot, which are recognized by fusing three different modalities such as speech command (using a CNN), hand motion (using a LSTM), and body motion. In an industrial setting instead, the actions of interest may include either general actions (e.g., walking, standing) or specific actions and gestures, depending on the main objective of the action recognition. For example, action recognition can be used to ensure human safety in HRC by monitoring what people are doing within the robot's workspace. In [5], authors monitor people moving near a robotic arm and recognize actions such as *passing*, *observing*, *dangerously observing*, *interacting*. The recognition system follows a multi-modal approach: a 3D-CNN extracts features from sequences of RGB frames, while signals from haptic sensors are used to detect collisions between humans and the robot using a 1D-CNN.

When tasks involve close collaboration between humans and robots, human action recognition can be used to enhance productivity by monitoring different stages of the collaboration. In this case, the set of actions to be recognized typically includes different operations that the human needs to fulfill for completing the overall assembly task. For example, in [17] the set of actions included *grab a tool* from a toolbox, *insert a screw*, *tight the screw*, *put back the tool* in the toolbox; authors proposed an action recognition classifier based on a CNN trained on a combination of skeleton features and signals from EMG and IMU sensors. In [6], similar activities were considered such as *taking a product or a component*, *move a product*, *grab a tool*, *put on screws*, *hold a product*, *tighten the screws*, *check product* and *place product*; in such case authors focused on hands information only, proposing a system that combines hands' pose and images cropped around the hands. In [27] the assembly actions include instead *cleaning*, *hammering*, *polishing*, *smearing*, *installing*, *screwing* and *marking*, all recognized using a two-stream CNN.

Despite sharing some common high-level actions and gestures, many of the human–robot collaboration systems presented in this section consider a very specific set of actions related to a particular task; in many cases, the datasets used for these systems are often not published or are too task-specific to be useful in other contexts. In addition, action recognition is generally addressed by focusing on either body or hand information. To the best of our knowledge no work has attempted to recognize actions using body and hands information together.

In this work we address both problems, namely the lack of a general dataset and the recognition of actions involving either the body or the hands. On one hand, we propose a general framework to recognize both body actions and hands gestures in a collaborative scenario; on the other hand the system is developed using a novel action recognition dataset acquired on purpose, including common actions of a collaborative task so as to be generalizable to various scenarios and applications.

3. Methods

In this section, we provide a detailed description of the main parts of our general framework for human action recognition in collaborative scenarios. The framework follows a skeleton-based action recognition approach, since skeletons provide a robust representation of the human movements free of any disturbances such as external objects and illumination; this is important especially for collaborative scenarios, where both the human and the robot are moving and the human worker should interact with many objects and tools. A schematic representation of the proposed system is shown in Fig. 1, highlighting the main steps involved. Our system takes as input a sequence of RGB-D frames and predicts the corresponding action performed according to a given set of actions of interest.

In the first stage, we estimate human poses in a sequence of RGB frames by means of 3D pose estimation methods. In particular, we focus on estimating the pose information of both the body and hands, since many collaborative gestures could be executed by using only the hands. Common approaches based on projection from 2D to 3D using depth images work well for body joints, but can be very inaccurate regarding hands joints, leading to invalid 3D coordinate values. To handle such problem, in the 3D pose estimation stage a "2D to 3D lifting" method is also considered, in order to estimate valid 3D joints when the 3D projection method fails. The estimated 3D joints are then fed to a set of graph convolutional HAR networks derived from Shift-GCN [30], a state-of-the-art action recognition architecture. Indeed, for the final action recognition stage, we rely on an ensemble of classifiers, each one trained to recognize actions from a different set of joints (e.g., body, left hand and right hand). The final output of the system is a weighted average of the output scores of all the classifiers, which according to our experiments in Section 5 proves to be more accurate and robust with respect to a single model trained on body and hands poses together.

3.1. 3D pose estimation

In a previous version of the proposed framework [7], the 3D pose estimation stage was mainly based on the projection from estimator output, the 2D pose, by means of depth information. Although this is a common approach to obtain 3D pose information of a person's body, it proved to be very inaccurate to obtain the 3D pose of the hands, leading to many incomplete 3D skeletons with missing hand joints. To address such limitation, in this work we investigate different alternative methods to compute a complete 3D skeleton composed of body and hands joints. In particular, we consider "2D to 3D lifting" and monocular 3D Human Shape estimation approaches to estimate valid 3D hand joints and alleviate the problem of incomplete 3D skeletons. In the following, each of these approaches is described in detail, highlighting their pros and cons.

3.1.1. 3D pose estimation based on 2D to 3D projection

For 2D pose estimation we rely on the OpenPose architecture [29], which provides different pretrained models for multi-person pose estimation, allowing to estimate in real-time either 15, 18 or 25 body keypoints, 42 hand keypoints and 70 face keypoints. The output of OpenPose is a 2D skeleton describing the pose of each person in the input image. From this information we compute a 3D skeleton by means

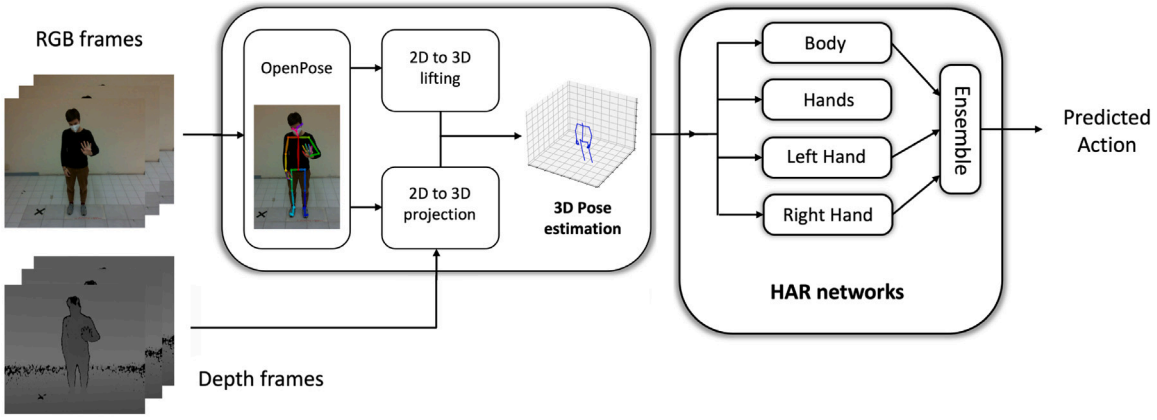


Fig. 1. Overview of the proposed action recognition system. In the first stages, 3D skeletons including body and hand joints are estimated from sequences of RGB-D frames by means of 3D pose estimation. In the last stage, the classification step, actions are predicted using an ensemble of skeleton-based action recognition models, each one trained to recognize actions from a particular set of 3D joints.

of projection, in order to have a more standard and general representation to be used as input for the action classifiers. In this work, we highlighted that this is not an optimal solution, but we still considered it since there are many applications exploiting direct projection. Given a calibrated RGB-D camera and its extrinsic parameters, it is possible to transform the depth image by aligning it to the RGB image, obtaining a direct mapping for each pixel; for each 2D keypoint (x_P, y_P) estimated by OpenPose in the input image, we use the information in the aligned depth image to compute the keypoints coordinates in the 3D space using re-projection and the intrinsics parameters of the camera. The depth information acquired with a RGB-D sensors is usually inaccurate around borders and for small objects, like the fingers of the hand in our case. Therefore, to improve accuracy and robustness of the 3D keypoints, we do not consider the raw depth value, but we take instead the median value in a 5×5 window centered at coordinates (x_P, y_P) in the depth image. Nevertheless, the 2D to 3D projection method is heavily affected by the depth information leading to inaccuracies and missing joints when the subject is far from the camera or partially occluded; this is especially true for the joints of the hands, which can be easily occluded during interactions with objects. To alleviate such problem, we consider a simplified skeletal representation composed of a subset of OpenPose joints by removing the joints which are missed more frequently; in particular, as shown in Fig. 2 our choice for the hand joints is limited to: the wrist, three joints each for the thumb, index and middle finger, and two joints for the ring finger; for a total of 12 joints for each hand. The rest of the joints are omitted, being in general more difficult to be estimated and less useful for the action recognition process. For what concerns the body, some keypoints such as the ones corresponding to eyes, ears and feet are not considered either, leading to a 15 joints body model comprising head, neck, shoulders, elbows, wrists, pelvis, hips, knees and ankles.

3.1.2. 3D pose estimation based on 2D to 3D lifting

Although many works have attempted to predict the 3D pose directly from just an RGB image, information obtained by 2D pose estimation is still important to achieve accurate 3D predictions about the human pose. For example, the MeTRAbs architecture [39] can predict a relative 3D pose from a single 3D image up to arbitrary translation; by also providing 2D pose predictions it is then possible to disambiguate the relative pose estimated and obtain the absolute pose. MeTRAbs takes in input a RGB image and computes the relative 3D predictions exploiting a volumetric heatmaps representation. The absolute pose is then computed through a differentiable reconstruction module, based on a linear least squares formulation derived from the pinhole camera model. The networks use a detector to isolate human instances, meaning that it can work with multiple people.

However, MeTRAbs was developed to predict only body joints, meaning that no information on hand joints is provided. In order to obtain a 3D skeleton complete of body and hands joints, we tried to pair MeTRAbs with another model capable of estimating the 3D pose of the hands such as InterNet [48], but from some early tests this model proved to be unreliable and unable to generalize to images other than those on which it was trained. We then focused on "2D to 3D lifting" approaches, based on a two-stage procedure that first translate the image into 2D human pose and then convert the 2D human pose to 3D human pose. Inspired by [42], where authors showed that using a transformer layer is beneficial for lifting 2D body keypoints, we replicate the same idea to lift hands keypoints provided by OpenPose from 2D to 3D. Our version is first trained on InterHand2.6 dataset [48], a large scale dataset containing images of hands from multiple point of view, using the ground-truth hand poses projected on the image coordinates as input to the transformer.

3.1.3. 3D pose and human shape estimation

Differently from the previous methods, SMPLify-X [44] aims to recover 3D pose by fitting a parametric model. This is the union of three state-of-the-art models for body, face and hands, namely SMPL [43], FLAME [46] and MANO [45]. The new model is named SMPL-X and is defined by the function $M(\beta, \theta, \psi) : R^{|\beta| \times |\theta| \times |\psi|} \rightarrow R^{3N}$, where $\theta \in R^{3(K+1)}$ contains the K pose parameters for body, hands and face, plus the parameters for the global rotation; the $\beta \in R^{|\beta|}$ the shape parameters and the $\psi \in R^{|\psi|}$ the expression parameters. SMPLify-X fits the SMPL-X model given only an RGB image and the 2D keypoints for body, hand and face, which can be easily estimated by means of OpenPose as discussed in Section 3.1.1. It minimizes the objective function:

$$E(\beta, \theta, \psi) = E_J + \lambda_{\theta_b} E_{\theta_b} + \lambda_{\theta_f} E_{\theta_f} + \lambda_{m_h} E_{m_h} + \lambda_\alpha E_\alpha + \lambda_\beta E_\beta + \lambda_\epsilon E_\epsilon + \lambda_C E_C \quad (1)$$

where E_{θ_f}, E_ϵ, E_{m_h}, E_β and E_{θ_b}, are face pose, expression, hand pose, shape and body pose priors, respectively. E_C avoids self interpenetration. E_α avoids extreme bending for elbow and knees. Finally, E_J is the distance between the 2D input joints and the estimated 3D points projected to the image. The body prior is computed by a variational autoencoder [49]. The λ are scalars to help optimization, which is performed with the limited-memory BFGS [50] optimizer.

In this work, we are not interested in the facial expression or the shape parameters and only the 3D joints locations for body and hands in the fitted model are considered. Note that the computation of the whole SMPL-X model is slow and could not be used in real case scenarios, such as human–robot collaboration; nevertheless we investigate also this kind of methods to obtain an interesting benchmark when compared with the other 3D pose estimation methods.

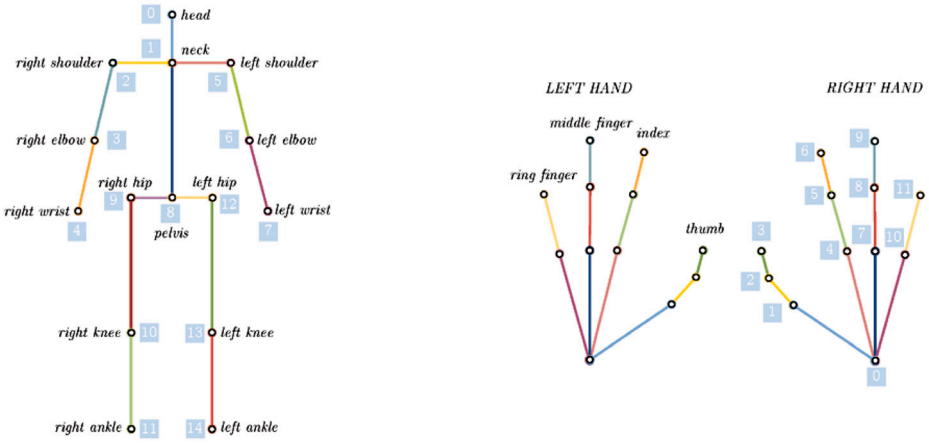


Fig. 2. Body and hand joints considered in the proposed framework, together with their associated ID. We consider a total of 15 joints describing the body, and 12 joints to describe each one of the hands.

3.2. Action and gesture recognition

The action recognition module in our framework is based on the Shift-GCN architecture [30], a graph convolutional network which achieved state-of-the-art performance on the NTU RGB+D dataset [33]. Compared to other architectures, Shift-GCN is more efficient and requires less computational power, due to the use of graph shift convolution operations. This was one of the main aspects that guided our choice, since in human–robot collaboration scenarios a quick recognition of human actions is essential to obtain a smooth and responsive collaboration.

The Shift-GCN input is a sequence of skeletons, (i.e., a sequence of 3D joints coordinates), and its architecture is composed of 10 blocks, each one including a spatial graph shift convolution operation and an adaptive temporal shift operation. The final output layer contains 60 nodes to match the number of actions in the NTU RGB+D dataset. As generally done for graph convolutional networks, the length of the skeleton sequence is a hyperparameter of the network used. Specifically, such types of networks require a fixed-length input such as a sequence of a predefined number of skeletons (e.g., a sequence of 125 skeletons corresponding to a 5-s video sequence with pose estimation at 25 FPS); if the sequence contains less skeleton than the required number, a padding strategy is used to add "skeletons" to the sequence.

Unlike the NTU dataset, where actions were classified based only on the body pose information, in this work we aim to develop a general framework capable of recognizing both actions and gestures which can occur during the collaboration between a human worker and a robot. In collaborative scenarios, many actions and gestures are limited to hand movements while the worker's body remains still (e.g., "stop" and "confirm" signals). To achieve this, we propose various action recognition networks that are based on the Shift-GCN architecture and are tailored to recognize specific collaborative actions from a specific set of joints. In particular, we consider the following set of joints:

- **wholebody**, which includes the 39 joints shown in Fig. 2 describing the pose of both the body and the hands;
- **body**, including the 15 joints which describe the pose of the body;
- **hands**, including the 24 joints which describe both hands together;
- **single hand**, including the 12 joints of a hand (i.e., wrist and fingers' joints).

Although the methods presented in Sections 3.1.2 and 3.1.3 are capable of estimating all the hand joints, the different Shift-GCN architectures have been developed using the subsets of joints identified in Section 3.1.1 to alleviate the missing joints problem of the 2D to 3d projection-based method, in order to have a direct comparison of the performance obtained by the various methods.

As in the original Shift-GCN architecture, the input sequences of skeleton joints go through some pre-processing steps before being fed into the network. These steps include translating the joint coordinate reference frame to a central joint of the skeleton and normalizing the joint coordinates. These operations let the network consider body movements with respect to the body, which makes the input more suitable for the network and easier to generalize to different scenarios. For our networks, we choose the neck joint as the origin of the new reference frame (i.e., Joint 1 in Fig. 2). The z-axis of the new reference frame is taken parallel to the segment connecting the pelvis (i.e., Joint 8) and neck joints, while the x-axis is considered parallel to the segment connecting the shoulder joints (i.e., Joints 2 and 5). For the networks that consider only the hand joints, we use the same convention for the reference frame's axes, but placing its origin on the wrist joint (i.e., Joint 0) of each hand, or the right hand wrist when both hands are considered. This allows the network to express hand movements with respect to a local reference frame while maintaining their relative orientation with respect to the rest of the body.

3.3. Ensemble averaging of the classifiers predictions

As the last step of our proposed framework, all the outputs of the action recognition models are combined together by means of ensemble techniques to compute the final prediction. Ensemble is a common technique in the machine learning field that combines several base models in order to produce one optimal predictive model, improving the overall accuracy and robustness.

We propose two main ensemble approaches: an ensemble of the *body* and *hands* models, and an ensemble of the *body* model with both single hand models (i.e., *left_hand* and *right_hand* models). In both approaches, the information is combined at the score-level, namely the output of the *softmax* activation function in the last layer of the networks. Considering for example the first approach (i.e., $body + hand$ models), we compute the final score as a weighted sum of the score of each model. The predicted action l_{pred} is then obtained by taking the argmax of the final score,

$$l_{pred} = arg\,max\left(\alpha_b\,\mathbf{o}_b + \alpha_h\,\mathbf{o}_h\right), \quad \text{with } \alpha_b + \alpha_h = 1 \tag{2}$$

where N is the number of actions of interest, $\mathbf{o}_b \in [0,1]^N$ is the output score of the *body* network, $\mathbf{o}_h \in [0,1]^N$ is the output score of the *hands* network and α_o, α_h are the corresponding weights.

In the second approach (i.e., *body* + *left_hand* + *right_hand* models) we take into account also the fact that some actions or gestures can be performed using only one hand (e.g., confirm, left, right, stop) while

Table 1
Analysis of the most common actions and gestures considered in the literature for human–robot collaboration applications.

Action	Works	Gesture	Works
Walk	[5,13,51]	Stop	[4,51–55]
Rest	[51,56,57]	Ok/Confirm	[4,54,55]
Pick	[6,17,51,52,54,56–60]	Up	[4,54,55]
Place	[6,52,54,60–62]	Down	[4,54,55]
Screw	[6,27,58,60,61,63]	Forward	[55]
Insert/Join	[17,61,62]	Backward	[53,55]
Hammer	[27,56,57,63]	Left	[4,54,55]
Hand To	[17,51,59,64]	Right	[4,54,55]
Require	[61,64]	Point	[51]

the other one remains still or even not visible. The final score is still computed as a weighed sum of the models' score, but considering a weight $\alpha_{ih} = 0$ if the hand $i, i \in \{left, right\}$ is the only one not visible or in a rest position. We assume in this case that the hand whose action is labeled as "rest" falls in a group of actions in which only one of the two hands is actively moving, while the other one stands still; the actual action is therefore related to the moving hand and the other one is irrelevant in terms of action recognition.

4. IAS-lab collaborative HAR dataset

As highlighted in Section 2, a variety of actions are commonly used in human–robot collaborative applications. In order to create a general framework that fits many different collaborative settings, we trained our models on a set of actions and gestures that are representative of those commonly found in the literature. For example, many works include actions such as *"Grab a tool"*, *"Pick a piece"* or *"Pick the hammer"* which can be generalized as a *"Pick"* action if we only focus on the human movements; the specific object to be picked can then be identified using a dedicated object detector.

With this in mind, we investigated in the literature what types of human actions are most common in human–robot collaboration applications, focusing only on the type of movement while ignoring any specific tool or object. The result of this analysis is shown in Table 1, which highlights how the main actions and gestures considered in the literature for collaborative scenarios can be described by means of 9 actions and 9 gestures: actions mainly include movements of the whole body or some parts (e.g., arms) that occur in collaborative assembly tasks, whereas gestures include many commands performed with the hands to give feedback to the robot.

The set of actions shown in the Table 1 thus represents a general set of actions suitable for many human–robot collaboration applications. We used this set of actions as the basis for our dataset, introducing a further classification into four main categories, as reported in Table 2. The first category includes general movements that a person can make within the robot's workspace, such as *Walk* which includes all movements of a worker moving around the workspace and *Rest* indicating that the human operator is not working. The second category includes the most common assembly actions performed by the human worker, whereas the third category includes all signals that the human can use to request or pass objects to the robot during collaboration. Finally, the fourth category includes all gestures used to communicate instructions to the robot, including directions of movements and signals of confirmation or halting.

The set of selected actions was chosen to be generic, and can generally be distinguished without the need to pair them with specific objects or tools. It can be observed that many of the actions involve an active use of the hands and that hand movements can be a significant factor in recognizing the action being performed, particularly for gestures and collaborative actions.

Based on the set of actions reported in Table 2, we collected a new dataset of people performing such actions in our laboratory. Six different participants were asked to perform each of the 18 actions selected

Table 2
Actions and gestures in the IAS-Lab Collaborative HAR dataset.

Group	Actions
Spatial movements	Walk, Rest
Assembly actions	Pick, Place, Screw, Insert/Join, Hammer
Collaborative actions	Hand To, Require
Communication gestures	Stop, Ok/Confirm, Up, Down, Forward, Backward, Left, Right, Point

5 times, resulting in a total of 540 samples. Each sample is a sequence of RGB-D frames lasting approximately 5 s, recorded using an Intel Realsense L515 camera. The camera was placed at a distance of around 2.5 m from the subjects to capture the entire body during all actions. Some samples from the acquired dataset are illustrated in Fig. 3. During the collection of the dataset, the subjects were only provided with the name of the action to be performed, without receiving any additional guidance on how to execute it. This approach increased the variability of the dataset as subjects performed the same actions in different ways as shown in Fig. 4. The goal is to recognize actions and gestures that are performed as naturally as possible, without creating a rigid set of movements and having to provide detailed instructions to users. This ensures easier, more immediate, and natural communication between people and robots.

5. Experimental results on IAS-lab dataset

The evaluation of the proposed Human-Action Recognition (HAR) framework was conducted using the IAS-Lab Collaborative HAR dataset. However, this dataset is not large enough for training deep learning models from scratch, as mentioned in Section 4. Therefore, the action recognition models were first trained on a larger dataset, such as the NTU RGB+D dataset, and then fine-tuned using the IAS-Lab Collaborative dataset. This approach allows for the utilization of the larger dataset to learn various features related to human movements, which can then be specialized for the human–robot collaboration scenario through fine-tuning. All the models discussed in the following sections were based on the official Shift-GCN architecture,[2] and trained using the hyper-parameters suggested by the Shift-GCN authors, using a NVIDIA® Titan RTX 2080 GPU.

5.1. Pre-training on the NTU RGB+D dataset

The NTU RGB+D dataset [33] is a large collection of RGB-D frames with pose annotations for multiple individuals performing 60 different actions, recorded using a multi-camera setup. The original dataset includes pose annotations with 25 joints that describe body poses, but they do not include hand joints. Moreover, the joints in the skeleton model are slightly different from those estimated by the OpenPose architecture used in our framework. Therefore, to make use of both body and hand pose information in our framework, we recreated the pose annotations for the NTU RGB+D dataset using the OpenPose network to predict both body and hand joints. For each RGB frame in the NTU RGB+D dataset, we run the OpenPose pose estimator to predict the 2D poses of the people in the images. Then, by using the associated depth frame, the corresponding 3D pose is computed by means of re-projection as described in Section 3.

The authors of the NTU RGB+D dataset proposed two benchmarks, one where subjects are split into training and testing groups (cross-subject benchmark), and another where data from different cameras are used as train and test data (cross-view benchmark). However, in this work, the primary objective is to use the NTU dataset to train models on a large collection of actions; for this reason, we did not

[2] https://github.com/kchengiva/Shift-GCN.

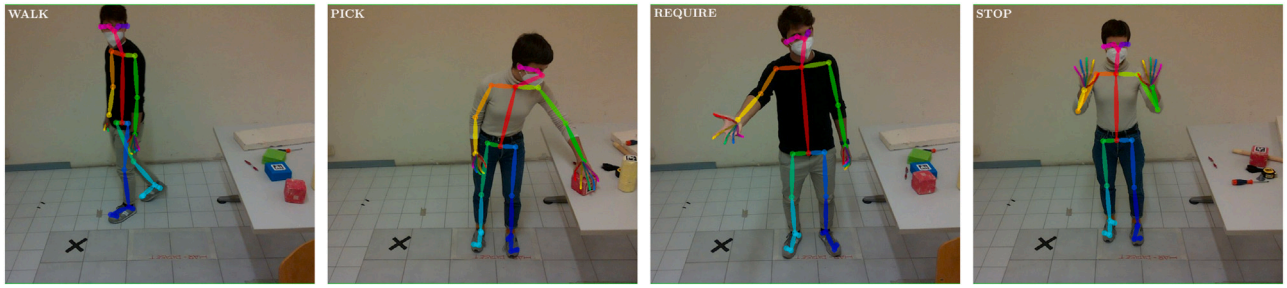


Fig. 3. Some samples from the IAS-Lab Collaborative HAR dataset, together with the skeletons outputs estimated dataset by means of OpenPose architecture.

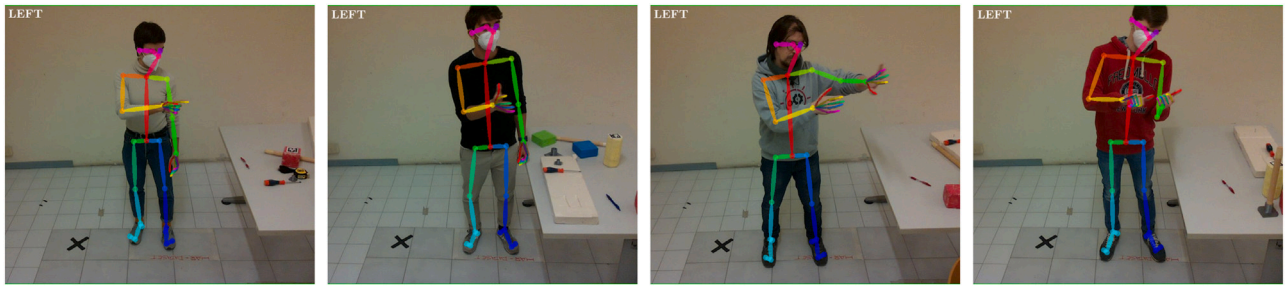


Fig. 4. Examples of variability in the IAS-Lab Collaborative HAR dataset. Each subject performed the requested action (e.g. *Left*) in a different manner.

Table 3
Experimental results on the NTU RGB+D dataset for each proposed model. Results are provided in terms of accuracy using the skeleton sequences extracted with OpenPose.

Model	#Joints	Top1%	Top5%
wholebody	39	59.44	81.34
body	15	**90.40**	**97.68**
hands	24	36.25	67.63
left hand	12	21.12	49.38
right hand	12	31.90	62.68

follow any of the proposed divisions but used it as a training set with the highest number of images. In particular, we considered all the data from cameras 2 and 3 as the training set and also added the data from camera 1, which is reserved for training in the original NTU RGB+D cross-subject benchmark. The remaining data, namely the test data in the cross-subject benchmark acquired with camera 1, was used as a validation set to monitor the training and prevent overfitting.

Using the dataset division described above, the models outlined in Section 3 were trained on sequences of 3D skeletons obtained from OpenPose outputs, considering the subset of the most important joints shown in Fig. 2. Specifically, a separate model was trained for each set of joints e.g., *body* joints, *left hand* and *right hand* joints) and a *whole_body* model considering all the available joints (i.e., *body* and *hand* joints). Results for each trained model on the NTU RGB+D dataset are presented in Table 3. The models were evaluated in terms of accuracy, utilizing both *Top1 accuracy* and *Top5 accuracy* metrics. The former represents the percentage of correctly predicted actions in the test set, while the latter is the percentage of actions whose correct prediction falls within the five highest softmax scores estimated by the network.

As shown in Table 3, the highest accuracy was achieved by the model trained only on the 15 selected body joints, while models trained on the hand joints had a low accuracy. This outcome was somewhat expected given that the NTU RGB+D dataset includes a wide range of daily actions (e.g., *drinking*, *eating*, *reading*) that involve minimal use of the hands. Many of these actions are primarily differentiated by body posture, and the hands provide only minimal information that is not enough for a model trained only on hand joints to distinguish between such a wide range of actions.

Even the *wholebody* model trained on both body and hand joints has a lower accuracy than the model trained on only body information. This suggests that including hand information may even harm the model, causing it to misinterpret more actions than when using body information alone. Out of the 39 input joints, only 15 describe the body pose, while more than the half represent hand information that does not provide enough knowledge to recognize actions.

It is worth noting that Shift-GCN architecture achieved a Top1 accuracy of 96.5% on the original NTU RGB+D dataset with body pose annotations, while our "body" model performed slightly worse with a Top1 accuracy of 90.4%. A direct comparison between the two results is not possible due to the use of slightly different train and test sets. However, the decrease in performance may be partly due to the new pose annotations and how the network handles partial inputs. If some of the required joints are missing from the input, the entire input skeleton is discarded. This occurred on several occasions when using the skeletons estimated by OpenPose (especially for hands when not clearly visible or partially occluded by objects), which caused entire sequences to be discarded when too many skeletons were missing.

5.2. Fine-tuning on the IAS-lab collaborative HAR dataset

Using the NTU RGB+D dataset, several action recognition models were trained to classify a wide range of daily activities. The large size of the dataset enabled the models to learn various low-level and mid-level features that can be useful also for action recognition in collaborative scenarios. All the models were fine-tuned on the IAS-Lab Collaborative HAR dataset described in Section 4, by changing the final layer of 60 nodes to a layer of 18 nodes to match the size of the new set of actions. To preserve the low-level and mid-level features learnt, all weights of the layers were frozen except for the last ones during fine-tuning. Specifically, denoting with $\ell i, i \in [1, 10]$ the 10 blocks of the Shift-GCN architecture, all the blocks were frozen except the final ones reported in Table 4 for each model. For models that achieved low accuracy on the NTU RGB+D dataset (e.g., hand models), more blocks were allowed to be retrained as low accuracy suggests that poor mid- and high-level features were learned, and more weights should be updated.

When fine-tuning the models on the IAS-Lab dataset, a cross-subject benchmark was applied, where the first five subjects were used for

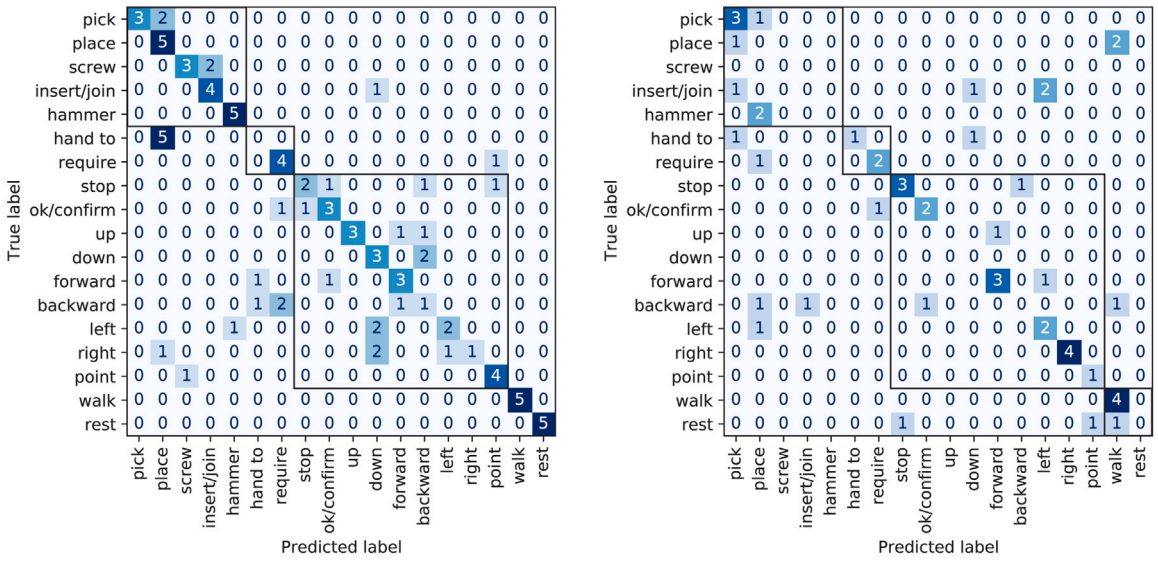


Fig. 5. Confusion matrices for the models fine-tuned on the IAS-Lab Collaborative HAR dataset. On the left, the confusion matrix for the *body* model. On the right, the confusion matrix for the *hands* model.

Table 4
Experimental results on the IAS-Lab Collaborative HAR dataset for each proposed model. Results are provided in terms of accuracy using the models pre-trained on the NTU RGB+D dataset.

Model	#Joints	Valid inputs	Top1%	Top3%	Learnable blocks
wholebody	39	45%	44.00	54.00	$\ell 7$, $\ell 8$, $\ell 9$, $\ell 10$
body	15	100%	**62.22**	**86.67**	$\ell 9$, $\ell 10$
hands	24	45%	50.00	64.00	$\ell 8$, $\ell 9$, $\ell 10$
left hand	12	75%	59.42	73.91	$\ell 6$, $\ell 7$, $\ell 8$, $\ell 9$, $\ell 10$
right hand	12	70%	59.42	71.01	$\ell 6$, $\ell 7$, $\ell 8$, $\ell 9$, $\ell 10$

training and the sixth subject was used for validation. The fine-tuned models were evaluated in terms of Top1 and Top3 accuracy and the results are presented in Table 4. Given the smaller number of actions in this case compared to the NTU dataset, the Top3 accuracy was chosen instead of the Top5 accuracy used in the previous case.

As seen in Table 4, the best results were still obtained with the model trained on body joints, which achieved a Top1 accuracy of 62.22%. Notably, the models trained on hand joints performed better on the IAS-Lab dataset compared to the NTU dataset, especially for models that consider each hand separately. In many sequences of the IAS-Lab dataset, the subjects performed actions with one hand while the other hand was at rest. These situations are ambiguous for a model that recognizes actions using information from both hands, resulting in a reduction of its overall accuracy.

However, none of the models were able to correctly classify all actions in the test set data, which highlights the complexity of the recognition task assigned. As mentioned in Section 4, the IAS-Lab dataset includes a variety of typical actions in human–robot collaboration scenarios, including general actions (e.g., *pick*, *place*) and hand gestures (e.g., *confirm*, *stop*). General actions are large movements that involve many body parts, such as walking or hammering. Hand gestures, on the other hand, are frequently used to communicate with the robot and involve only movements of the worker's hands while the rest of the body remains mostly still. Both the *body* model and the *single hand* models are highly accurate in classifying only one type of action, but perform poorly in the other one due to the lack of information. For example, as illustrated in Fig. 5, the *body* model accurately predicts actions such as *place*, *hammer* or *walk*, but has difficulty recognizing all gestures based on hand movements (e.g., *hand to*).

Intuitively, a model trained on both body and hands information should be the best one to recognize all the given actions and gestures, but in our experiments we found the opposite to be true, with the *wholebody* model achieving the lowest accuracy among the results reported in Table 4. This is mainly due to the fact that in several sequences the skeletons found were not complete with all joints, resulting in these sequences being discarded from the model's training set, and thus limiting the model ability to learn to recognize all actions of interest. The percentage of valid training sequences available for each model is reported in Table 4, which highlights how the available data was particularly limited for training the *wholebody* model.

5.3. Comparing different 3D pose estimation approaches

As noted in the previous section, 2D to 3D projection methods are a good approach for estimating 3D body joints but show many limitations when dealing with hand joints as well. Indeed, in many cases it is not possible to estimate the correct 3D position of hand joints due to invalid depth values or occlusions (e.g., a hand-held object), leading to incomplete skeletons and reducing the amount of data useful for training and testing the action classification networks.

This problem led us to investigate alternative methods for obtaining a 3D pose, capable of predicting a correct and complete skeleton of all joints even if the depth information is not reliable. In particular, we considered two main approaches: "2D to 3D lifting" and 3D human shape estimation. Both approaches infer the 3D pose from the output of a 2D pose estimator. In the former case the 3D pose is obtained by means of regression, while in the latter case the 3D pose is obtained by fitting a parametric human model on the 2D pose. For each of these methods, we first estimated the corresponding 3D poses on the IASLAB Dataset and then trained a Shift-GCN-based classifier on each subset of joints considered in the previous experiments. This allows a fairer comparison of the performance of the classifiers based on the new 3D pose results with the previous results obtained using the 2D to 3D projection method. Moreover, since in the previous experiment the presence of many incomplete skeletons caused several sequences to be discarded from the training and test set, in comparing the performance of the classifiers when varying the 3D pose estimation method two different evaluation settings were considered: in Table 5 all the classifiers are evaluated on the subset of valid test sequences where the 2D to 3D projection method computes enough complete 3D skeletons, while in Table 6 the classifiers are evaluated on all the IASLAB dataset test sequences.

Table 5
Experimental results on the IAS-Lab Collaborative HAR dataset for each proposed model on the subset of common test sequences. Results are provided in terms of accuracy using the models pre-trained on the NTU RGB+D dataset.

Method	Body		Hands		Left hand		Right hand		Wholebody	
	Top1	Top3	Top1	Top3	Top1	Top3	Top1	Top3	Top1	Top3
OpenPose 3D	62.22	86.67	50.00	64.00	59.42	73.91	59.42	71.01	44.00	54.00
Metrabs + 2D lift	**67.78**	83.33	69.39	79.59	60.87	72.46	66.18	76.47	63.27	81.63
SMPLify-x	55.56	71.11	**71.43**	85.71	72.46	84.06	**67.65**	75.00	53.06	67.35
OP + 2D lift	65.56	91.11	61.22	83.67	63.77	76.81	64.71	77.94	**69.39**	89.80
OP + SMPLify-x	**67.78**	87.78	67.35	87.76	**73.91**	79.71	64.71	79.41	**69.39**	91.84
Test sequences	90		49		69		68		49	

Table 6
Experimental results on the IAS-Lab Collaborative HAR dataset for each proposed model on all test sequences. Results are provided in terms of accuracy using the models pre-trained on the NTU RGB+D dataset.

Method	Body		Hands		Left hand		Right hand		Wholebody	
	Top1	Top3	Top1	Top3	Top1	Top3	Top1	Top3	Top1	Top3
OpenPose 3D	62.22	86.67	30.00	35.56	51.11	56.67	51.11	62.22	24.44	33.33
Metrabs + 2D lift	65.56	91.11	65.56	84.44	52.22	67.78	64.44	77.78	64.44	83.33
SMPLify-x	55.56	71.11	**70.00**	87.78	**63.33**	78.89	**66.67**	76.67	57.78	75.56
OP + 2D lift	**67.78**	83.33	68.89	87.78	55.56	72.22	64.44	78.89	68.89	86.67
OP + SMPLify-x	**67.78**	87.78	**70.00**	87.78	**63.33**	74.44	**66.67**	80.00	**71.11**	88.89
Test sequences	90		90		90		90		90	

In both tables the results of the following 3D pose estimation methods are reported: 2D to 3D projection from 2D poses estimated by OpenPose (*Openpose3D*); 2D to 3D lifting using Metrabs for the body and our version of [42] for the hands (*Metrabs+2Dlift*); 3D human shape estimation by means of parametric models (*SMPLify-x*). Since for body joints the result of projection from 2D to 3D still gives an accurate result, we also considered "hybrid" methods, where we keep *Openpose3D* body joints and integrate missing hand joints using one of the other methods: in *OP+2Dlift* hand joints are provided by lifting openPose 2D predictions to 3D using our version of [42], while in *OP+SMPLify-x* hand joints are provided by the parametric model fitted on OpenPose 2D joints by SMPLify-X.

As shown in Table 5, the action classifiers trained on the 3D poses obtained by the new methods considered outperform the previous results achieved using 2D to 3D projection (*Openpose3D*). Indeed, by being able to always predict a complete 3D skeleton (i.e., both body and hands joints), with 3D pose obtained by such methods, it is possible to train action classifiers for each body part on a larger amount of data improving the ability to then generalize over test sequences.

This observation is further confirmed by the results shown in Table 6, where the various action classifiers are evaluated on the whole test set. In this case, *Openpose3D* has a performance drop due to sequences with incomplete skeletons, especially in the case of *hands* and *wholebody* action classifiers; in contrast, there are no major differences in the performance of the other methods, an indication that the various action classifiers obtained with the new 3D poses are robust across the entire test set.

In summary, the introduction of alternative methods for 3D pose estimation has alleviated one of the main problems related to missing 3D hand joints, allowing for more robust action classifiers. In particular, the *SMPLify-x* method is the one that achieves the best results regarding hands, while for body joints it proves to be less robust than the *Openpose3D* method. Indeed, by investigating the 3D skeletons estimated by *SMPLify-x*, it can be seen that on several occasions it predicts an incorrect pose for the body (e.g., one leg forward instead of being backward), while the pose of the hands is always consistent with the real pose. This is confirmed in the results of method *OP+SMPLify-x*, where the body joints estimated by *SMPLify-x* are replaced with those obtained by means of 2D to 3D projection, which in fact obtains the best results for each subset of joints.

The time required to predict a 3D pose using *SMPLify-x* is about 50 s per frame, which makes this method difficult to apply in scenarios where real-time performance is required. Given such requirement, as in a human–robot collaboration scenario, a better solution is the *OP+2Dlift* method that in Table 6 achieves the second best results on all joint configurations; in that case, the 3D body joints are computed by means of 2D to 3D projection, while the 3D hand joints are estimated by lifting the 2D joints provided by Openpose.

5.4. Ensemble results using body and hands models

Body posture and hand posture are complementary information that, if properly combined, can greatly improve action recognition. As highlighted in previous experiments, combining this information when training a network has generally proven to be of little use or even inefficient: on the one hand, more sequences are required to learn the various relationships that may exist between body and hands during the actions of interest; on the other hand, the majority of hand joints over body joints may lead networks to focus more on hands than body information.

For these reasons, we developed our action recognition system by combining information at the score level by means of ensemble techniques: we run in parallel the models fine-tuned on the IAS-Lab dataset, and combine together their score predictions (i.e. the predicted probability for each actions) by means of a weighted sum. In particular, we investigated two main approaches: an ensemble of the *body* and *hands* models, and an ensemble of the *body* model with the models for each hand considered separately. In the former case we found that best results were obtained when weighting equally the contribution of the two hands, that is, using weights $\alpha_b = \alpha_h = 0.5$ in Eq. (2). Instead, when using a separate model for each hand we found that less importance should be given to the scores of the *left_hand* model, probably due to the fact that the majority of the subjects in the dataset were right-handed and tended use their right hand to perform the requested actions; the final weights we selected for this approach are $\alpha_b = \alpha_{rh} = 0.358$ and $\alpha_{lh} = 0.284$.

The results obtained using the two ensemble strategies proposed are reported in Table 7. In general, both strategies lead to an improvement of the Top1 accuracy with respect to the previous results, showing how in general combining body and hands information helps classifying actions. The best results were obtained with the ensemble of the *body* model with the single models for each hand, achieving a good improvement with respect to the results obtained by each model in Table 6. The major improvement is for the *Openpose3D* method,

Table 7
Experimental results on the IAS-Lab Collaborative HAR dataset using different ensembles of the fine-tuned models.

Method	Body + Hands		Body + Single hands	
	Top1	Top3	Top1	Top3
OpenPose 3D	65.56	87.78	66.67	90.00
Metrabs + 2D lift	65.56	91.11	67.78	88.89
SMPLify-x	66.67	85.56	70.00	84.44
OP + 2D lift	68.89	90.00	**75.56**	88.89
OP + SMPLify-x	**71.11**	93.33	**75.56**	91.11

Table 8
Inference time in milliseconds for each proposed model considering different GPU hardware.

Hardware	Body	Hands	Left hand	Right hand	Wholebody
NVIDIA GeForce 2080	6.18	6.28	6.14	6.22	6.43
NVIDIA GeForce 3060	3.45	3.41	3.40	3.40	3.61

where 3D skeletons are obtained by means of 2D to 3D projection. In this case, the main advantage of the ensemble strategy is to help alleviate the missing joints problem: a sequence is considered a valid input if at least the body or a hand is detected, reducing the number of overall frames and sequences discarded. But even in the case where all skeletons are complete with all joints, the "body + single hands" ensemble strategy allows a better action classification. In particular, *OP+2Dlift* and *OP+SMPLify-x* methods achieve the best result on the IASLAB dataset, with a respective accuracy improvement of +7% and +4% with respect to the corresponding *wholebody* action classifier.

5.5. Run-time analysis

In this section we provide an analysis of the execution time of the main module of the proposed system. In particular, the modules requiring most of the computational time can be identified in the 3D pose estimation module and the ensemble of graph convolutional networks. For the 3D pose estimation module, different variations has been considered in the previous experiments in Tables 5 and 6. OpenPose3D can predict 3D body and hands joints with an approximate speed of 25 FPS on a NVIDIA GeForce 2080 GPU, where the computational time is almost entirely due to the 2D pose estimator (i.e., OpenPose) since the 3D projection step using depth images is irrelevant in terms of execution time. The "Metrabs+2D lift" is composed of two main parts: Metrabs body pose estimator and a 2D lifting method to estimate 3D hand joints: Metrabs runs an approximate speed of 25 FPS on a NVIDIA GeForce 2080 GPU, while the 2D lifting method computes 3D hand joints from 2D hand joints of OpenPose with an execution time of 95 FPS on a NVIDIA GeForce 2080-Ti. Finally, SMPLify-x allows to estimate an accurate 3D skeleton composed of body and hand joints, but its execution time is about 50 s per frame. Regarding the ensemble of graph convolutional networks, an analysis of the inference time of each network is reported in Table 8 for different GPU models. The only difference between the networks considered is the number of skeleton joints considered in the input sequence, as described in Table 3. As shown in Table 8, the networks are all capable of running at a very high frame rate on today's very common hardware (e.g., NVIDIA GeForce 2080 and NVIDIA GeForce 3060). Moreover, all models occupy just under 2 GiB of GPU memory, and it is therefore possible to run the different models needed for the ensemble (e.g., body, lefthand and righthand models) in parallel on the same midrange GPU.

6. Experimental results on HRI30

To prove the robustness and the generalization capabilities of our action recognition framework, a series of experiments has been performed also on the HRI30 dataset [13], proposed for action recognition in industrial human–robot interaction scenarios. The HRI30 dataset contains 30 categories of industrial-like actions like *Pick Up Drill* or *Walking with Polisher*. For each category 98 video clips have been collected, for a total of 2940 video clips; each video clip has been recorded with a Realsense D435i camera framing a large working area, and then manually cut to contain only the performed action.

The HRI30 dataset has several differences from the IASLAB dataset considered in the previous sections: it contains only RGB information, with only one person per frame but viewed in profile and at a much greater distance from that considered in the IASLAB dataset acquisition. An example of the scene differences from the two datasets is shown in Fig. 6. Moreover, the action categories to be recognized in the HRI30 are generally composed of both human movements (e.g., *Move Backwards*, *Move Right*) and objects (e.g., *Drill*, *Polisher*), while the IASLAB dataset addresses general collaborative actions which only depends on human movements.

In this section, our framework is evaluated on the HRI30 dataset to demonstrate how it can easily generalize on different scenarios, even with only RGB data and a very different viewpoint. Moreover, we will show that by coupling the proposed framework with an object classifier it is possible to predict all the HRI30 action classes outperforming existing methods.

6.1. Generalization on the HRI30 dataset

Our proposed framework relies on skeleton-based action classifiers, with 3D skeletons providing a robust representation of the human movements free of any disturbances such as external objects and illumination. This allows our system to learn actions classification from joints movements alone, thus generalizable to different scenarios. On the contrary, methods for action recognition based on RGB images or video analysis struggle with new scenes, since they are strictly dependent on training data. A proof to the hypothesis can be inferred from the HRI30 dataset. As shown in Fig. 6, this dataset presents a very different scenario from that found in the IASLAB dataset, constituting an interesting test case on which validating generalization capabilities.

Since the IASLAB dataset has been built considering general actions which could occur in an human–robot collaboration, the 30 categories proposed in the HRI30 dataset can be easily mapped in the IASLAB dataset actions. In particular, most of the HRI30 categories describe a *Move* action in some direction, which can be mapped into a *Walk* action considering the IASLAB dataset set of classes reported in Table 2; the remaining classes in the HRI30 dataset have a direct counterpart in the IASLAB dataset, namely *Pick*, *Place*, *Hand To* for the *Deliver* category and *Rest* for the *No Collaborative* category. With such mapping, we evaluate the model trained on the IASLAB dataset directly on the HRI30 dataset, in order to demonstrate how the skeletal representation allows to easily generalize on a new scenario. Note that the HRI30 dataset does not contain depth information but only RGB data, so only monocular or lifting models, namely *Metrabs+2Dlift* and *SMPLify-x*, were tested. The results of this validation are shown in Table 9 together with state-of-the-art action classifier based on video analysis. The table highlighting how our skeleton-based action classifiers maintain good performance when applied to different collaborative scenarios.

6.2. Evaluation on the HRI30 dataset

Our proposed framework has been designed to be general and directly usable in various scenarios for recognizing a general set of common collaborative actions. Such a framework, thus, constitutes a large knowledge base that can be further specialized on a more specific set of human actions to deal with specific tasks and applications. Consider, for example, the HRI30 dataset, which includes many collaborative actions related to movements within the working area such as *"Move Backwards While Drilling"*, *"Move Backwards While Polishing"* or *"Move Diagonally Forwards Left with Drill"*. According to the analysis made in Section 4, all

Fig. 6. A comparison between the IAS-Lab Collaborative HAR dataset (left) and the HRI30 dataset (right).

Table 9
Experimental results of the models trained on the IASLAB dataset when tested on the HRI30 dataset. All reported inference times are obtained as the average inference time of the networks on each test sequence of the IASLAB dataset on a Nvidia Titan RTX 2080 GPU.

Method	IASLAB		HRI30		Inference
	Top1	Top3	Top1	Top5	time [s]
SlowOnly [20]	64.44	81.11	29.05	52.98	1.13
C2D [65]	46.67	74.44	25.00	41.19	0.15
I3D [65]	47.78	81.11	18.81	32.23	0.22
VideoSwin [66]	64.44	86.67	10.36	32.26	0.59
TimeSformer [67]	54.44	78.89	18.21	51.79	0.14
STGCN++ [68]	58.89	83.33	48.81	32.23	0.02
Ours (Metrabs + 2Dlift)	67.78	88.89	**84.12**	90.48	0.008
Ours (SMPLfy-x)	**70.00**	84.44	80.95	85.71	0.008

these actions fall under the general action category *Walk* which can be easily predicted by our framework as shown in Section 6.1. However, in a particular application it may be important to recognize not only that a person is walking but also the direction in which he/she is moving. This motivates the need to have a framework capable of recognizing general actions, but at the same time easy to be specialized on a specific set of actions related to a given task.

The 30 action categories included in the HRI30 dataset include various combinations of 3 main objects (i.e., Drill, Polisher, Object) and 14 main body actions: *"Deliver"*, *"MoveBackwards"*, *"MoveDiagonallyBackwardLeft"*, *"MoveDiagonallyBackwardRight"*, *"MoveDiagonallyForwardLeft"*, *"MoveDiagonallyForwardRight"*, *"MoveForward"*, *"MoveLeft"*, *"MoveRight"*, *"NoCollaborative"*, *"PickUp"*, *"PutDown"*, *"Using"* and *"Walking"*.

Using the official train/test splits provided in the HRI30 dataset [13], we fine-tuned our action classifiers on such 14 body actions. In particular, we fine-tuned the action classifier based on "2D to 3D lifting" pose estimation (i.e., *Metrabs+2Dlift*), which does not rely on depth information and provides in general more accurate results for actions based on body joints. For fine-tuning we kept the hyper-parameters suggested by the Shift-GCN authors [30], changing the number of nodes in the final layer and lowering the initial learning rate to 0.01 so as not to change the previous training too much; all the training procedures have been performed using a NVIDIA® Titan RTX 2080 GPU. Results have been reported in Table 10, which shows a very good performance in terms of Top1 and Top5 accuracy for the fine-tuned action classifier. This result demonstrates the possibility and ease of specializing our framework in human–robot collaboration scenarios with a different set of action of interests.

Recognizing people's actions from a skeletal representation allows our framework to be very robust in recognition and easily generalizable to different scenarios. However, by discarding the RGB information the action recognition framework has no way to differentiate any objects with which the person interacts. When the actions to be recognized also include objects, as in the case of HRI30 dataset, it is then possible to extend the framework by coupling it with an object detector or classifier. For example, we trained an object classifier derived from [69] to recognize the 3 objects included in the HRI30 categories by analyzing images cropped around the human body. In particular, we extracted the person 2D pose for the frames in the train set of each split; given such 2D pose, we then computed a bounding box that encapsulates the whole person and cropped the image around such bounding box to build a training set for the object classifier. The object classifier has been trained considering also a 4th class corresponding to a *No object* to handle cases in which the person is picking or placing the object. The performance of this object classifier are reported in Table 10 for each HRI30 split, showing that in almost all the cases it is able to correctly recognize the object handled by the person.

Coupling together the action and the object classifiers, it is possible to predict the original categories included in the HRI30 dataset. We combine the predictions of the two classifiers at the score-level (i.e., the output of the last softmax layer), by computing their joint probability under the assumption that action and object are independent events:

$$Z = P(action, object) = P(action) \cdot P(object)$$

Table 11 shows the results obtained by our action framework and object classifier in terms of Top1 and Top5 accuracy for each split. In the same Table we also reported the state-of-the-art methods evaluated on the HRI30 dataset by the respective authors. As marked in bold, our approach outperforms all the other methods achieving the best performance in terms of Top1 accuracy on all the splits.

7. Conclusions

In this work, we propose a unified framework for action recognition in human–robot collaboration scenarios. Our framework is based on skeleton-based action classifiers which can recognize various body movements and hand gestures commonly used in collaborative tasks, making it a versatile solution for various real-world applications. Different 3D pose estimation methods have been considered and investigated to develop the proposed system, since the quality and completeness of the 3D pose plays a crucial role in the action classifiers performance. Our experiments showed that 2D to 3D lifting methods provide a more robust 3D pose than 2D to 3D projection methods commonly used in the literature. The system has been evaluated on a novel dataset including general actions of human–robot collaboration scenarios, which could be used as a benchmark to further drive research in this field. Our experiments showed that using an ensemble of action classifiers, each trained to recognize actions from different joints, has several benefits: it improves the overall accuracy and makes the system more robust to possible missing joints in the estimated 3D skeletons. Considering also the HRI30 dataset, we demonstrate that the proposed framework can be easily specialized on more detailed human actions and achieve state-of-the-art results when coupled with an object detector or classifier. Some limitations of the proposed system are the fixed length of the input sequence, and the fact that the ensemble classifier assumes as

Table 10
Experimental results on the HRI30 dataset, considering classifiers for human movements and objects separately.

Classifiers	Split 1			Split 2			Split 3		
	Top1	Top3	Top5	Top1	Top3	Top5	Top1	Top3	Top5
Actions	88.33	99.05	99.76	87.62	99.84	99.84	87.52	98.95	99.81
Objects	97.74	99.05	100	97.30	98.89	100	98.00	99.05	100

Table 11
Experimental results on the HRI30 dataset with respect to the 30 action classes.

Method	Split 1		Split 2		Split 3	
	Top1	Top5	Top1	Top5	Top1	Top5
SlowOnly [20]	86.55	99.76	83.49	99.84	82.43	**99.90**
TSN [21]	74.05	97.20	73.98	99.05	73.71	98.86
IRCSN [22]	79.17	**99.88**	74.64	99.84	77.67	99.62
VideoSwin [66]	82.62	**99.88**	81.90	**100.00**	82.67	99.81
TimeSformer [67]	61.90	98.33	72.70	97.62	63.24	98.38
Ours	**88.10**	98.81	**87.30**	98.99	**88.00**	99.05

the subject's dominant hand the right hand. As future development of the system we would like to remove this assumption and make the system capable of inferring the subject's dominant hand thorough the analysis of several consecutive actions of a same subject during a complex assembly task; in this scenario, the weights α_{rh} and α_{lh} in the ensemble module could be adjusted periodically over time based on a statistic of recognized left- or right-handed actions (e.g., using Bayes' rule to update the belief about the dominant hand in the light of the obtained statistic). To address the limitation with fixed size input sequences, we will consider the use of warping techniques to be more robust to the same actions done at different speeds (and thus different durations). As future research directions, we also plan to evaluate the framework in a real human–robot collaboration task to monitor human movements during an assembly process. Moreover, we will further investigate the robustness of the framework to different viewpoints considering a multi-camera setup.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Matteo Terreran reports financial support was provided by European commission.

Data availability

Data will be made available on request.

Acknowledgments

The research leading to these results has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101006732 (DrapeBot).

References

[1] V. Villani, F. Pini, F. Leali, C. Secchi, Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications, Mechatronics 55 (2018) 248–266.
[2] E. Matheson, R. Minto, E.G. Zampieri, M. Faccio, G. Rosati, Human–robot collaboration in manufacturing applications: a review, Robotics 8 (4) (2019) 100.
[3] W. Kim, L. Peternel, M. Lorenzini, J. Babič, A. Ajoudani, A human-robot collaboration framework for improving ergonomics during dexterous operation of power tools, Robot. Comput.-Integr. Manuf. 68 (2021) 102084.
[4] H. Liu, T. Fang, T. Zhou, L. Wang, Towards robust human-robot collaborative manufacturing: Multimodal fusion, IEEE Access 6 (2018) 74762–74771.
[5] F. Mohammadi Amin, M. Rezayati, H.W. van de Venn, H. Karimpour, A mixed-perception approach for safe human–robot collaboration in industrial automation, Sensors 20 (21) (2020) 6347.
[6] T. Kobayashi, Y. Aoki, S. Shimizu, K. Kusano, S. Okumura, Fine-grained action recognition in assembly work scenes by drawing attention to the hands, in: 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2019, pp. 440–446.
[7] M. Terreran, M. Lazzaretto, S. Ghidoni, Skeleton-based action and gesture recognition for human-robot collaboration, in: International Conference on Intelligent Autonomous Systems, Springer, 2022, pp. 29–45.
[8] Y. Jiang, C. Cao, X. Zhu, Y. Ma, Q. Cao, RGBD-based real-time 3D human pose estimation for fitness assessment, in: 2020 3rd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), IEEE, 2020, pp. 103–108.
[9] A. Malaguti, M. Carraro, M. Guidolin, L. Tagliapietra, E. Menegatti, S. Ghidoni, Real-time tracking-by-detection of human motion in RGB-D camera networks, in: 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), IEEE, 2019, pp. 3198–3204.
[10] F. Lygerakis, A.C. Tsitos, M. Dagioglou, F. Makedon, V. Karkaletsis, Evaluation of 3D markerless pose estimation accuracy using openpose and depth information from a single RGB-d camera, in: Proceedings of the 13th ACM International Conference on PErvasive Technologies Related To Assistive Environments, 2020, pp. 1–6.
[11] Y. Cai, L. Ge, J. Liu, J. Cai, T.-J. Cham, J. Yuan, N.M. Thalmann, Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2272–2281.
[12] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, M.J. Black, Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image, in: European Conference on Computer Vision, Springer, 2016, pp. 561–578.
[13] F. Iodice, E. De Momi, A. Ajoudani, HRI30: An action recognition dataset for industrial human-robot interaction, in: 2022 26th International Conference on Pattern Recognition (ICPR), 2022, pp. 4941–4947, http://dx.doi.org/10.1109/ICPR56361.2022.9956300.
[14] K. Liu, M. Zhu, H. Fu, H. Ma, T.-S. Chua, Enhancing anomaly detection in surveillance videos with transfer learning from action recognition, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 4664–4668.
[15] A. Prati, C. Shan, K.I.-K. Wang, Sensors, vision and networks: From video surveillance to activity recognition and health monitoring, J. Ambient Intell. Smart Environ. 11 (1) (2019) 5–22.
[16] C.M. Ranieri, S. MacLeod, M. Dragone, P.A. Vargas, R.A.F. Romero, Activity recognition for ambient assisted living with videos, inertial units and ambient sensors, Sensors 21 (3) (2021) 768.
[17] M. Al-Amin, W. Tao, D. Doell, R. Lingard, Z. Yin, M.C. Leu, R. Qin, Action recognition in manufacturing assembly using multimodal sensor fusion, Procedia Manuf. 39 (2019) 158–167.
[18] W. Bo, M. Fuqi, J. Rong, L. Peng, D. Xuzhu, Skeleton-based violation action recognition method for safety supervision in the operation field of distribution network based on graph convolutional network, CSEE J. Power Energy Syst. (2021).
[19] C. Chen, R. Jafari, N. 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), IEEE, 2015, pp. 168–172.
[20] C. Feichtenhofer, H. Fan, J. Malik, K. He, SlowFast networks for video recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.
[21] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. Van Gool, Temporal segment networks: Towards good practices for deep action recognition, in: European Conference on Computer Vision, Springer, 2016, pp. 20–36.
[22] D. Tran, H. Wang, L. Torresani, M. Feiszli, Video classification with channel-separated convolutional networks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 5552–5561.
[23] J. Yu, H. Gao, W. Yang, Y. Jiang, W. Chin, N. Kubota, Z. Ju, A discriminative deep model with feature fusion and temporal attention for human action recognition, IEEE Access 8 (2020) 43243–43255.
[24] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, S.W. Baik, Action recognition in video sequences using deep bi-directional LSTM with CNN features, IEEE Access 6 (2017) 1155–1166.

[25] C. Feichtenhofer, X3d: Expanding architectures for efficient video recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 203–213.
[26] X. Wen, H. Chen, Q. Hong, Human assembly task recognition in human-robot collaboration based on 3D CNN, in: 2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), IEEE, 2019, pp. 1230–1234.
[27] Q. Xiong, J. Zhang, P. Wang, D. Liu, R.X. Gao, Transferable two-stream convolutional neural network for human action recognition, J. Manuf. Syst. 56 (2020) 605–614.
[28] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, 2014, arXiv preprint arXiv:1406.2199.
[29] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, Y. Sheikh, OpenPose: realtime multi-person 2D pose estimation using part affinity fields, IEEE Trans. Pattern Anal. Mach. Intell. 43 (1) (2019) 172–186.
[30] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, H. Lu, Skeleton-based action recognition with shift graph convolutional network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.
[31] Z. Liu, H. Zhang, Z. Chen, Z. Wang, W. Ouyang, Disentangling and unifying graph convolutions for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 143–152.
[32] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, W. Hu, Channel-wise topology refinement graph convolution for skeleton-based action recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.
[33] A. Shahroudy, J. Liu, T.-T. Ng, G. Wang, Ntu rgb+ d: A large scale dataset for 3d human activity analysis, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1010–1019.
[34] J. Wang, X. Nie, Y. Xia, Y. Wu, S.-C. Zhu, Cross-view action modeling, learning and recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2649–2656.
[35] D.T. Tran, H. Yamazoe, J.-H. Lee, Multi-scale affined-HOF and dimension selection for view-unconstrained action recognition, Appl. Intell. 50 (2020) 1468–1486.
[36] M. Liu, H. Liu, C. Chen, Enhanced skeleton visualization for view invariant human action recognition, Pattern Recognit. 68 (2017) 346–362.
[37] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, N. Zheng, View adaptive neural networks for high performance skeleton-based human action recognition, IEEE Trans. Pattern Anal. Mach. Intell. 41 (8) (2019) 1963–1978.
[38] D. Maji, S. Nagori, M. Mathew, D. Poddar, YOLO-pose: Enhancing YOLO for multi person pose estimation using object keypoint similarity loss, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2637–2646.
[39] I. Sárándi, T. Linder, K.O. Arras, B. Leibe, Metrabs: metric-scale truncation-robust heatmaps for absolute 3d human pose estimation, IEEE Trans. Biom. Behav. Identity Sci. 3 (1) (2020) 16–30.
[40] M.R.I. Hossain, J.J. Little, Exploiting temporal information for 3d human pose estimation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 68–84.
[41] D. Pavllo, C. Feichtenhofer, D. Grangier, M. Auli, 3D human pose estimation in video with temporal convolutions and semi-supervised training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 7753–7762.
[42] W. Li, H. Liu, R. Ding, M. Liu, P. Wang, W. Yang, Exploiting temporal contexts with strided transformer for 3d human pose estimation, IEEE Trans. Multimed. (2022).
[43] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, M.J. Black, SMPL: A skinned multi-person linear model, ACM Trans. Graph. (TOG) 34 (6) (2015) 1–16.
[44] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A.A. Osman, D. Tzionas, M.J. Black, Expressive body capture: 3d hands, face, and body from a single image, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10975–10985.
[45] J. Romero, D. Tzionas, M.J. Black, Embodied hands: Modeling and capturing hands and bodies together, ACM Trans. Graph. 36 (6) (2017).
[46] T. Li, T. Bolkart, M.J. Black, H. Li, J. Romero, Learning a model of facial shape and expression from 4D scans, ACM Trans. Graph. 36 (6) (2017) 1–17.
[47] G.S. Martins, L. Santos, J. Dias, The GrowMeUp project and the applicability of action recognition techniques, in: Third Workshop on Recognition and Action for Scene Understanding (REACTS). Ruiz de Aloza, 2015.
[48] G. Moon, S.-I. Yu, H. Wen, T. Shiratori, K.M. Lee, Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image, in: European Conference on Computer Vision, Springer, 2020, pp. 548–564.
[49] D.P. Kingma, M. Welling, Auto-encoding variational bayes, 2013, arXiv preprint arXiv:1312.6114.
[50] J. Nocedal, S.J. Wright, Nonlinear equations, in: Numerical Optimization, Springer, 2006, pp. 270–302.
[51] A. Roitberg, A. Perzylo, N. Somani, M. Giuliani, M. Rickert, A. Knoll, Human activity recognition in the context of industrial human-robot interaction, in: Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific, IEEE, 2014, pp. 1–10.
[52] K. Zhang, W. Xu, B. Yao, Z. Ji, Y. Hu, H. Feng, Human motion recognition for industrial human-robot collaboration based on a novel skeleton descriptor, in: 2020 IEEE 16th International Conference on Automation Science and Engineering (CASE), IEEE, 2020, pp. 404–410.
[53] Z. Song, Z. Yin, Z. Yuan, C. Zhang, W. Chi, Y. Ling, S. Zhang, Attention-oriented action recognition for real-time human-robot interaction, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 7087–7094.
[54] S. Sheikholeslami, A. Moon, E.A. Croft, Cooperative gestures for industry: Exploring the efficacy of robot hand configurations in expression of instructional gestures for human–robot interaction, Int. J. Robot. Res. 36 (5–7) (2017) 699–720.
[55] P. Tsarouchi, A.-S. Matthaiakis, S. Makris, G. Chryssolouris, On a human-robot collaboration in an assembly cell, Int. J. Comput. Integr. Manuf. 30 (6) (2017) 580–589.
[56] W. Tao, Z.-H. Lai, M.C. Leu, Z. Yin, R. Qin, A self-aware and active-guiding training & assistant system for worker-centered intelligent manufacturing, Manuf. Lett. 21 (2019) 45–49.
[57] W. Tao, M.C. Leu, Z. Yin, Multi-modal recognition of worker activity for human-centered intelligent manufacturing, Eng. Appl. Artif. Intell. 95 (2020) 103868.
[58] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, J. Chen, Deep learning-based human motion prediction considering context awareness for human-robot collaboration in manufacturing, Procedia CIRP 83 (2019) 272–278.
[59] A. Ghadirzadeh, X. Chen, W. Yin, Z. Yi, M. Björkman, D. Kragic, Human-centered collaborative robots with deep reinforcement learning, IEEE Robot. Autom. Lett. 6 (2) (2020) 566–571.
[60] P. Wang, H. Liu, L. Wang, R.X. Gao, Deep learning-based human motion recognition for predictive context-aware human-robot collaboration, CIRP Ann. 67 (1) (2018) 17–20.
[61] E. Coupeté, F. Moutarde, S. Manitsaris, Multi-users online recognition of technical gestures for natural human–robot collaboration in manufacturing, Auton. Robots 43 (6) (2019) 1309–1325.
[62] W. Tao, M. Al-Amin, H. Chen, M.C. Leu, Z. Yin, R. Qin, Real-time assembly operation recognition with fog computing and transfer learning for human-centered intelligent manufacturing, Procedia Manuf. 48 (2020) 926–931.
[63] C. Chen, T. Wang, D. Li, J. Hong, Repetitive assembly action recognition based on object detection and pose estimation, J. Manuf. Syst. 55 (2020) 325–333.
[64] M. Melchiorre, L.S. Scimmi, S. Mauro, S.P. Pastorelli, Vision-based control architecture for human–robot hand-over applications, Asian J. Control 23 (1) (2021) 105–117.
[65] X. Wang, R. Girshick, A. Gupta, K. He, Non-local neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7794–7803.
[66] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, H. Hu, Video swin transformer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 3202–3211.
[67] G. Bertasius, H. Wang, L. Torresani, Is space-time attention all you need for video understanding? in: ICML, Vol. 2, 2021, p. 4.
[68] H. Duan, J. Wang, K. Chen, D. Lin, Pyskl: Towards good practices for skeleton action recognition, in: Proceedings of the 30th ACM International Conference on Multimedia, 2022, pp. 7351–7354.
[69] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

Matteo Terreran is a postdoctoral researcher at the University of Padova, Department of Information Engineering. He received his MSc degree in Automation Engineering from the University of Padova, Italy in 2017, and the Ph.D. degree in Information Technology in 2021. His main research interests involve multiple topics related to computer vision and robotics, including intelligent perception for human-robot collaboration, human body pose estimation and human action recognition.

Leonardo Barcellona received the master's degree in Computer Engineering from the University of Padova, Italy in 2021. He is currently pursuing Ph.D. degree in Artificial Intelligence with the Intelligent and Autonomous Systems Laboratory, University of Padova, Italy and Politecnico di Torino, Torino, Italy. His research interest include human pose estimation, semantic segmentation, computer vision, robotics, artificial intelligence and deep learning.

Stefano Ghidoni is Full Professor at the University of Padova, Department of Information Engineering. His main research interests involve multiple topics related to computer vision for robotics, including intelligent perception for human-robot collaboration, semantic segmentation and intelligent video surveillance. He is also active in the field of pattern recognition for medical image analysis. Stefano Ghidoni received his MSc degree in Telecommunication Engineering from the University of Parma, Italy, in April, 2004, and the Ph.D. degree in Information Technology in 2008.