

# Sociable and Ergonomic Human-Robot Collaboration through Action Recognition and Augmented Hierarchical Quadratic Programming

Francesco Tassi<sup>\*1,2</sup>, Francesco Iodice<sup>\*1,2</sup>, Elena De Momi<sup>2</sup>, and Arash Ajoudani<sup>1</sup>

**Abstract**—The recognition of actions performed by humans and the anticipation of their intentions are important enablers to yield sociable and successful collaboration in human-robot teams. Meanwhile, robots should have the capacity to deal with multiple objectives and constraints, arising from the collaborative task or the human. In this regard, we propose vision techniques to perform human action recognition and image classification, which are integrated into an Augmented Hierarchical Quadratic Programming (AHQP) scheme to hierarchically optimize the robot’s reactive behavior and human ergonomics. The proposed framework allows one to intuitively command the robot in space while a task is being executed. The experiments confirm increased human ergonomics and usability, which are fundamental parameters for reducing musculoskeletal diseases and increasing trust in automation.

## I. INTRODUCTION

The number of collaborative robots (Cobots) in human-populated environments has increased exponentially due to their human-comforting properties. Thanks to strength, rigidity, and precision, Cobots have led to achieve significant developments in Human-Robot Collaboration (HRC), allowing for greater profitability and redistribution of work, thus reducing Musculoskeletal Disorders (MSDs) [1].

This collaboration, however, poses multiple open challenges. For example, it is essential for a Cobot to perceive the intentions of the human and distinguish between a collaborative or non-collaborative behaviour. When collaborating, the robot has to recognize human actions and understand intentions since the operator might interact through the use of multiple tools and objects that the robot should recognize to facilitate the current task. In addition, HRC is usually conceived within a shared workspace that is fixed in the work environment (e.g., on a workbench), whereas it might be of interest to allow the operator to move the shared workspace while continuing with the task (e.g., reach the next workstation while interacting, reducing cycle times).

Recently, action recognition has become an important topic, not only for computer vision but also for HRC. Most of the works are based on video processing through deep learning techniques [2], [3]. However, production-related scenarios tend to be oversimplified, and potential complexities due to, e.g., dirty background or similarity between

objects or movements, remained largely unexplored. Against these limitations, we use the SlowFast [4] network based on a 3D ResNet [5] model, which runs on a dataset of action sequences to improve the recognition accuracy.

However, action recognition alone does not ensure higher performances when it is applied to the robot under real circumstances. Indeed, to the authors’ knowledge, state-of-the-art methods [2], [3], [6] solve this problem independently from robot control. Our goal is to provide a synergetic framework that fluently integrates action recognition at the control level. To do so, we require an appropriate control logic capable of fully exploiting Cobots redundancy. This is possible by defining multiple priority levels of a stack of tasks [7]. Current studies on hierarchical control, range from humanoids [8], and teleoperation [9] to surgery [10]. Hierarchical Quadratic Programming (HQP) [11] solves multiple Quadratic Programming (QP) problems, establishing strict non-conflicting priorities [12].

In HRC, however, it might be beneficial to adjust the desired End Effector’s (EE) trajectories online, based on external interactions. To address this, we proposed in [13] an Augmented-HQP (AHQP) scheme for the augmented Inverse Kinematics (IK) problem. This is based on the creation of a Human-Robot Shared Workspace (HRSW), which is the feasible Cartesian region defined by QP constraints, in which HRC can occur. Inside it, thanks to the augmented scheme, the robot can optimally and autonomously adjust both its reference and actual trajectories (based on constraints and objectives). Thanks to the AHQP, we then introduced in [14] human ergonomics, which we exploit in this work to further integrate human intentions and actions recognition.

We take a large step from our previous works, by considering a moving and continuous interaction point for HRC. Indeed, to the authors’ knowledge, all the studies addressing human ergonomics in HRC either employ fixed robots [15] or consider a precise interaction area (e.g., a workbench for assembly [16]). Here instead, we provide a HQP-based controller capable of following the human while optimizing ergonomics. In addition, the proposed method allows dealing with ergonomics efficiently at the control level, by avoiding computationally intensive human models running online, as mostly done in the literature [17], and not having to rely on noisy or cumbersome acquisitions of all human joints coordinates (i.e., via camera or motion capture hardware). Indeed, these methods often require the online solution of the human dynamics/kinematics, which significantly affects computation times. In [17], an optimal posture is found in human’s joint coordinates, which then needs to be processed

\* : Both authors contributed equally to this work

<sup>1</sup>HRI<sup>2</sup> Lab, Istituto Italiano di Tecnologia, 16163, Genova, Italy.  
francesco.tassi, francesco.iodice@iit.it

<sup>2</sup>Department of Electronics Information and Bioengineering, Politecnico di Milano, Milan, Italy.

This work was supported in part by the ERC-StG Ergo-Lean (Grant Agreement No. 850932), in part by the European Union’s Horizon 2020 research and innovation programme, Grant Agreement No. 871237 (SOPHIA).

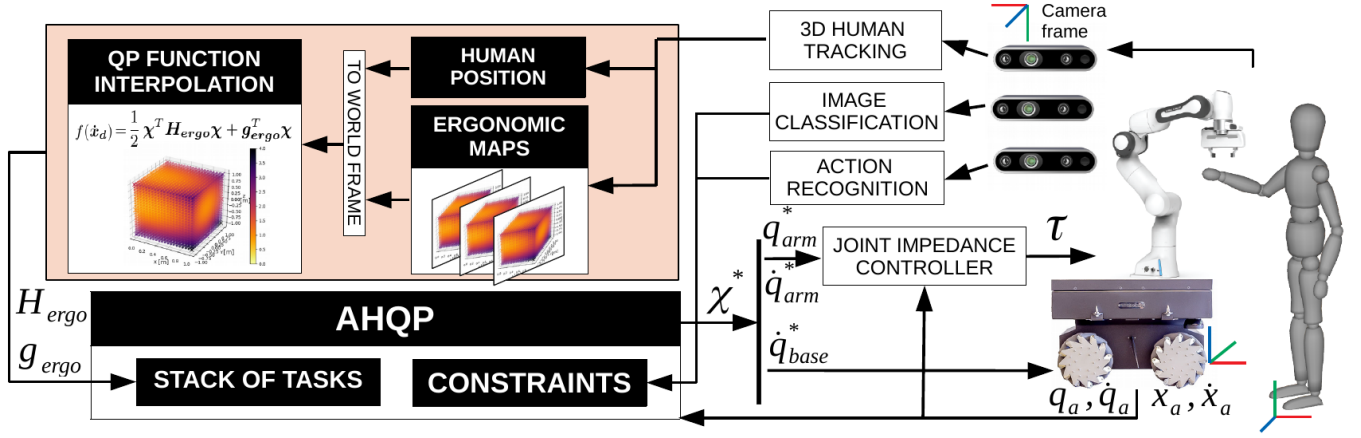


Fig. 1: Block scheme of the AHQP-based control framework proposed for HRC. The vision system allows to inform the controller about the current behaviour of the human, and to plan the successive tasks based on human intentions (linked to actions recognition). This in turn permits ergonomics optimization and successful task completion, thanks to the identification of the optimal solution  $\chi^*$ .

by both the motion planner and by the robot controller. Using the AHQP-based controller instead, only one step is required, reducing cycle times and, as importantly, hierarchically formulating multiple objectives and constraints.

Finally, we integrate a new perception module so that human intentions (e.g., collaborative and non-collaborative) and actions (e.g., walking directions, intended tool use, handover) can be detected and anticipated to create reactive robot responses. The studied scenario is adaptable to most industrial environments. In particular, first the human delivers a generic object to the Cobot, which is recognized thanks to an image classification module. An action recognition module then informs the Cobot about human intentions, in view of the manufacturing operations to be performed on the same object. The contributions of this work can be summarily listed as follows.

- A model for action recognition on constituent elements of a manufacturing sequence.
- A model for the binary classification on a customized dataset of objects' surfaces.
- Human action detection and intention recognition are integrated directly at control level through AHQP.
- With respect to [14], ergonomic HRC is introduced for moving interaction, and an HQP-based HRSW softening strategy is defined.

In what follows, we first describe the overall framework in Sec. II. We review in the first part of Sec. III the AHQP framework of [14], while we present the proposed control method in the second part. In Sec. IV, we describe the vision methods implemented. Finally, we validate the proposed structure through the experiments in Sec. V.

## II. OVERALL FRAMEWORK

The idea of the overall framework is to identify human actions and intentions during HRC, acting on the Cobot to accommodate them. Indeed, while most techniques consider a fixed HRC area, and require the operator to adapt to the robot, our goal is to improve the task's degree of flexibility, allowing the human to move in space (e.g., reach the next workstation, grab another tool or item for assembly), while

the robot follows the human and optimizes ergonomics. In the considered scenario, an object is first handed to the robot, identified by image classification and finally grasped. Successively, the operator can decide which tool to grab, to work on the object (i.e., drill, polisher), and the robot can recognize the intention via the action recognition module and act accordingly (i.e., properly orienting the object's surface). Furthermore, the robot uses the detection of additional human motions to interpret intents and follow the human in space, assisting him/her in completing the primary task. This will be detailed in Sec. V-D. Fig. 1 depicts the block scheme of the proposed framework, in which we employ three RGB-D cameras. The first is mounted on the mobile base to provide clear visibility of the EE. This allows to identify and correct the object's orientation during the activity. The other two are set at a distance that frames the whole working area, preventing occlusions to the greatest extent. One is used for action detection and the other for 3D human tracking, which is required to update online the position and orientation of the ergonomics map defined in Sec. III-B.

## III. ERGONOMICS-AWARE AHQP

### A. Augmented Kinematics

The classic hierarchical structure for  $k \in \{1, \dots, p\}$  priority levels, with decreasing task importance down to  $p$  writes as:

$$\begin{aligned} & \min_{\chi} \frac{1}{2} \|A_k \chi - b_k\|^2 \\ & s.t. \quad C_1 \chi \leq d_1, \dots, C_k \chi \leq d_k \\ & \quad E_1 \chi = f_1, \dots, E_k \chi = f_k, \end{aligned} \quad (1)$$

$\chi \in \mathbb{R}^s$  being the generic  $s$ -dimensional variable to optimize,  $n_{t_k}$  is the dimension of the  $k$ -th task, subject to  $n_{e_k}$  and  $n_{i_k}$  equality and inequality constraints respectively, through the matrices  $A_k \in \mathbb{R}^{n_{t_k} \times s}$ ,  $C_k \in \mathbb{R}^{n_{i_k} \times s}$ ,  $E_k \in \mathbb{R}^{n_{e_k} \times s}$  and vectors  $b_k \in \mathbb{R}^{n_{t_k}}$ ,  $d_k \in \mathbb{R}^{n_{i_k}}$ ,  $f_k \in \mathbb{R}^{n_{e_k}}$ . To establish the strict hierarchy between successive tasks, it is necessary that the task at a lower priority level  $k-1$  is projected in the nullspace of the task holding a higher priority  $k$ . This is achieved thanks to the condition  $A_{k-1} \chi = A_{k-1} \chi_{k-1}^*$  demonstrated in [12]. To account for this condition in (1), it is necessary to update the constraints matrices and vectors

at each step by formulating  $E_1 = 0, f_1 = 0$ , up to  $E_k = A_{k-1}, f_k = A_{k-1} \chi_{k-1}^*$ .

The IK problem of an  $n$ -DoF redundant robot is

$$\min_{\dot{\mathbf{q}}} \|\mathbf{J}\dot{\mathbf{q}} - \dot{\mathbf{x}}\|^2 \quad (2)$$

where  $\dot{\mathbf{q}} \in \mathbb{R}^n$  is the desired joint velocity,  $\dot{\mathbf{x}} \in \mathbb{R}^m$  the task space velocity and  $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{m \times n}$  is the task Jacobian matrix. The AHQP scheme proposed in [13] solves problem (2) through the augmented state  $\chi$ , with the desired Cartesian velocity  $\dot{\mathbf{x}}_d \in \mathbb{R}^m$  as part of the optimization variable:

$$\chi = \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{x}}_d \end{bmatrix}, \chi \in \mathbb{R}^{s=n+m}. \quad (3)$$

Using a Closed-loop IK (CLIK) scheme in augmented form, the augmented IK writes as in [14]:

$$\begin{aligned} \min_{\dot{\mathbf{q}}, \dot{\mathbf{x}}_d} \left\| \begin{bmatrix} \mathbf{J} & -(I + \mathbf{K}_p \Delta t) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{x}}_d \end{bmatrix} - \mathbf{K}_p(\mathbf{x}_{d_{t-1}} - \mathbf{x}_a) \right\|^2 \\ = \min_{\chi} \|\mathbf{A}_{clik}\chi - \mathbf{b}_{clik}\|^2 \end{aligned} \quad (4)$$

where  $\mathbf{x}_a, \mathbf{x}_d \in \mathbb{R}^m$  are the actual and desired Cartesian poses of the EE respectively,  $\mathbf{K}_p \in \mathbb{R}^{m \times m}$  is the positive-definite diagonal gain matrix responsible for convergence of the error,  $\Delta t$  is the control period that is set to 1ms, and  $\mathbf{x}_{d_{t-1}}$  is the value of  $\mathbf{x}_d$  at the previous time instant. Eq. (4) complies with the structure of (1) and can be solved hierarchically through  $\mathbf{A}_{clik} \in \mathbb{R}^{m \times s}$  and  $\mathbf{b}_{clik} \in \mathbb{R}^m$ .

### B. Human Ergonomics

Based on our previous work [14], we hereby describe how to integrate human ergonomics into the hierarchical framework thanks to the augmentation process. In HRC, it is important to optimize human ergonomics through Cobots, not by merely acting on their reference trajectories to be followed. Indeed, by defining priorities between productivity and human features at the control level, an optimal trajectory can be generated. However, to achieve this in real-time, a lightweight formulation is difficult to achieve when considering the highly redundant human kinematics computation. Therefore, based on the assumptions made in [14], we identify an ergonomics map in Cartesian space, simplifying the ergonomics problem and allowing for its tractability in the AHQP framework.

To assess human ergonomics and identify the ergonomics map, a Rapid Entire Body Assessment (REBA) score is considered. Given its highly discrete nature, though, the different contributions that compose the overall score (arms, trunk, legs, and neck) are considered separately and ultimately averaged. This provides continuous variations, useful to distinguish between similar postures that would correspond to the same REBA score. Dealing with HRC, it is obvious to assume  $\mathbf{x}_h = \mathbf{x}_d$ , for which the coordinates of the human hand  $\mathbf{x}_h$  correspond to the desired EE position, meaning both human and robot are ideally collaborating. This leads to the explicit formulation of the ergonomics score with respect to a subset of the optimization variable as  $e_s = f(\mathbf{x}_d)$ , where the score  $e_s \in \mathbb{R}$  can be employed in the HQP.

The other assumption is related to the definition of the ergonomics map. In particular, we assume that two persons (with the same kinematic parameters, center of mass position, and orientation in space) reach for a specific point in space with very similar postures of the overall kinematic chain  $\mathbf{q}_h$ , over multiple repetitions [14]. Indeed, despite the high degree of redundancy, this was confirmed by observing multiple subjects and recording their postures, noticing that they all tend to assume a configuration that minimizes the overall human joints activation. This, in turn, provides a measure of overall ergonomics associated with each point in the reachable space. Ultimately, the optimal ergonomics task is written as a function of  $\dot{\mathbf{x}}_d$  and in QP form as:

$$\min_{\dot{\mathbf{q}}, \dot{\mathbf{x}}_d} e_s = \min_{\dot{\mathbf{x}}_d} f(\dot{\mathbf{x}}_d) = \min_{\chi} f_a(\chi) \quad (5)$$

$$= \min_{\chi} \frac{1}{2} \chi^T \mathbf{H}_{ergo} \chi + \mathbf{g}_{ergo}^T \chi \quad (6)$$

with  $\mathbf{H}_{ergo} \in \mathbb{R}^{s \times s}$ ,  $\mathbf{g}_{ergo} \in \mathbb{R}^s$  containing the parameters obtained from the mapping. Finally, this is used in the AHQP structure (1), to regulate the position in which the robot's EE has to collaborate with the human for the task.

### C. Softening of the augmented problem

This section describes the constraints regulating the augmented problem and defines a slack variable useful for constraints softening, necessary to achieve the advantages related to the HRSW. This control scheme provides the possibility of obtaining  $\dot{\mathbf{x}}_d$  as output from the optimization, avoiding the restrictions of a fixed reference trajectory provided by upstream planning. Thereby, it is possible to simply act on the Cartesian constraints that identify the HRSW  $\mathbf{x}_{d_{min}} \leq \mathbf{x}_d \leq \mathbf{x}_{d_{max}}$ , and the controller will provide the optimal solution based on the hierarchy. To achieve this, however, it is necessary to avoid the infeasibility problems that would arise when the HRSW is modified, and the current value of  $\dot{\mathbf{x}}_d$  lies outside. Besides, it is beneficial to provide a degree of flexibility with which the optimization variable can remain outside of its boundaries, at the expense of another task ranked higher in the hierarchy (e.g., optimal ergonomics). Therefore, a slack variable  $s$  is added on  $\mathbf{x}_d$ :

$$\begin{aligned} \min_{\chi, s} \frac{1}{2} \|s\|^2 \\ \text{s.t. } \mathbf{x}_{d_{min}} - s \leq \mathbf{x}_d \leq \mathbf{x}_{d_{max}} + s, \end{aligned} \quad (7)$$

softening the HRSW and avoiding infeasibility. Rewriting as:

$$\begin{aligned} \begin{bmatrix} \mathbf{0}_{m \times n} & \Delta t \mathbf{I}_{m \times m} & -\mathbf{I}_{m \times m} \\ \mathbf{0}_{m \times n} & -\Delta t \mathbf{I}_{m \times m} & -\mathbf{I}_{m \times m} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\mathbf{x}}_d \\ s \end{bmatrix} \leq \begin{bmatrix} \mathbf{x}_{d_{max}} - \mathbf{x}_{d_{t-1}} \\ \mathbf{x}_{d_{min}} - \mathbf{x}_{d_{t-1}} \end{bmatrix} \\ = \mathbf{C}_s \begin{bmatrix} \chi \\ s \end{bmatrix} \leq \mathbf{d}_s, \end{aligned} \quad (8)$$

where  $\mathbf{C}_s \in \mathbb{R}^{2m \times s+m}$ ,  $\mathbf{d}_s \in \mathbb{R}^{2m}$  and the state is further augmented by  $s$ . Ultimately, as we show in the experiments, we will update the HRSW online to make the robot follow a continuous trajectory dictated by the human.

All the other constraints on  $\dot{x}_d, \ddot{x}_d$  and  $q, \dot{q}, \ddot{q}$ , are solved through the inequality parameters  $C_k, d_k$  in (1) as:

$$\chi_{min} \leq \chi \leq \chi_{max}. \quad (9)$$

#### D. Person-specific Cartesian mapping

Considering a realistic HRC task, it is often necessary for the robot to interact with multiple people having various characteristics of height, posture, and overall kinematics. To address this, we extend the ergonomics behavior of Sec. III-B by defining multiple maps of the score tailored to subjects with different kinematics (i.e., one for tall, average, and short subjects, respectively). This further optimizes the ergonomics of the final robot's trajectory, which can now provide a response tailored to each operator.

#### E. Stack of tasks

The overall stack of tasks is composed of three strict hierarchical levels. Highest priority is given to the formulation of the CLIK defined in (4), while at the second level we minimize the slack variable  $s$  responsible for softening the Cartesian constraint on  $x_d$  (7). Lastly, we prioritize the ergonomics function (6) that is responsible for dictating the optimal  $x_d^*$ . This order, aims at the highest satisfaction of human ergonomics while remaining inside the HRSW.

The optimal solution  $\chi^*$  found, is composed of:

$$\chi^* = \begin{bmatrix} \dot{q}^* \\ \dot{x}_d^* \end{bmatrix} = \begin{bmatrix} \dot{q}_{base}^* \\ \dot{q}_{arm}^* \\ \dot{x}_d^* \end{bmatrix} = \begin{bmatrix} \dot{x}_{base}^* \\ \dot{q}_{arm}^* \\ \dot{x}_d^* \end{bmatrix} \quad (10)$$

where  $\dot{x}_{base}^* = (\dot{x}, \dot{y}, \dot{\theta})$  are the linear and angular velocities of the mobile base. While  $\dot{x}_{base}^*$  is sent to the velocity controlled base, for the manipulator a joint impedance controller is necessary to generate the desired control torques:

$$\tau = K_{q_d}(\dot{q}_{arm}^* - \dot{q}_a) + K_{q_p}(q_{arm}^* - q_a) + g(q_a) \quad (11)$$

where  $\dot{q}_a, q_a \in \mathbb{R}^n$  are the actual joint velocities and positions respectively,  $q_{arm}^*$  is obtained from numerical integration of  $\dot{q}_{arm}^*$ , whereas  $K_{q_p}, K_{q_d} \in \mathbb{R}^{n \times n}$  are the positive definite joint stiffness and damping matrices respectively, and  $g(q_a) \in \mathbb{R}^n$  is the gravity compensation term.

### IV. VISION ARCHITECTURES

#### A. Surface Classification Model

The surface classification model classifies the sides of the object shown in Fig. 2, used for the experiments in Sec. V. Below, we provide prerequisites and propose a method to use transfer learning, SVM, and ensemble classification.

1) *Two-Class Linear Support Vector Machine*: The linear support vector machine (SVM) is a supervised machine learning algorithm initially formulated as a binary discriminative classifier. It is trained using a set of labeled instances, called training data, defined as:  $\mathbf{S} = ((x_1, y_1), \dots, (x_N, y_N)) \subseteq (\mathbf{X} \times \mathbf{Y})^N$  where  $N$  is the number of instances,  $\mathbf{X} \subseteq \mathbb{R}^N$  is the input space and  $\mathbf{Y} = \{1, -1\} \subseteq \mathbb{R}$  is the output domain. Moreover, we refer to  $\mathbf{x}_i$  as instances and  $\mathbf{y}_i$  as their labels. The objective of SVM is to find the optimal hyperplane:

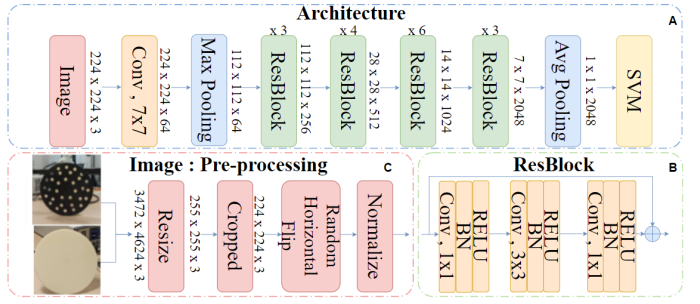


Fig. 2: A) Proposed architecture for surface classification based on fusion of ResNet-50 and SVM binary classifier via transfer learning via extractor features. B) Residual blocks with skip connection included in the network. C) Image pre-processing before training.

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \mathbf{x} + b = \sum_{j=1}^N w_j x_j + b. \quad (12)$$

SVM learns the parameters  $\mathbf{w}$  by solving the problem:

$$\begin{aligned} \min_{\mathbf{w}, \xi_i} \quad & \frac{1}{2} \mathbf{w}^T \mathbf{w} + \sum_{i=1}^N \xi_i \\ \text{s.t.} \quad & \xi_i \geq 0, \quad i = 1, \dots, N \\ & \mathbf{y}_i (\mathbf{w}^T \cdot \mathbf{x} + b) \geq 1 - \xi_i, \quad i = 1, \dots, N \end{aligned} \quad (13)$$

where  $\xi_i$  is a slack variable measuring the distance between the margin and the instances  $x_i$  lying on the wrong side of the margin,  $C$  is the trade-off parameter added to maximize the margin and minimize the classification error. The SVM solves the unconstrained optimization problem as follows:

$$\min_{\mathbf{w}} \quad \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N \max(1 - \mathbf{y}_i (\mathbf{w}^T \cdot \mathbf{x}_i + b), 0) \quad (14)$$

which is known as the L1-SVM primal problem with standard hinged loss. Its differentiable counterpart

$$\min_{\mathbf{w}} \quad \frac{1}{2} \mathbf{w}^T \mathbf{w} + C \sum_{i=1}^N \max(1 - \mathbf{y}_i (\mathbf{w}^T \cdot \mathbf{x}_i + b), 0)^2 \quad (15)$$

is known as L2-SVM. It minimizes a higher (quadratic) loss for points that violate the margin.

2) *Resnet*: ResNet [18] is one of the most powerful deep neural networks, whose versions (e.g., Resnet-18, Resnet-34, Resnet-50, etc.) only differ in the number of layers. The central idea of ResNet is the introduction of a so-called "skip connection". This method causes the residual mapping to allow the stacked layers to fit more easily than the desired background mapping. Also, the stacking of layers does not degrade the network's performance since the original information is not altered and flows directly from layer to layer throughout the network, preventing gradients from disappearing or exploding, improving learning and accuracy.

3) *Dataset*: The dataset includes 2000 images divided into two classes representing the surface of an object (Smooth and Drilled) uniformly distributed and manually annotated. We collected these images using an RGB camera, looking for depth variation and background complexity.

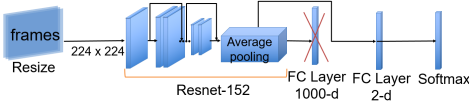


Fig. 3: Surface classification model with Resnet-152

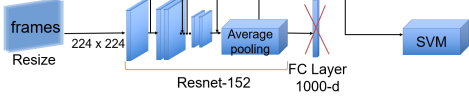


Fig. 4: Surface classification model with Resnet-152 + SVM

4) *Description Model*: Deep learning models usually need significant amounts of data and tuned architectures to predict with high accuracy. Being our data limited to 2000 samples, we investigate transfer learning techniques, where knowledge is transferred from a different domain. In particular, we use the Resnet network and the SVM binary classifier. The former, for its simple structure described in IV-A.2, provides better performances than other nets; the latter instead ensures high accuracy in handling small datasets. We choose the Resnet-50 architecture as the backbone for its high accuracy, computational speed, and ease of use, and we build a convolutional network (CNN), a combined model of ResNet-50 and SVM for ensemble classification, implementing transfer learning via extractor features. We use the pre-trained ResNet-50 weights on ImageNet as a starting point and freeze them to avoid their modification. This approach avoids back-propagation on the layers and significantly increases velocity during the training phases. Next, we remove the last fully connected layer having as output the number of previously towed classes on ImageNet, and use the features extracted from the average pooling layer to feed a binary SVM classifier (Fig. 2).

### B. Action Recognition Model

1) *Dataset*: The dataset used to train the network is called HRI30 [19]. It includes thirty classes associated with actions commonly performed in a collaborative industrial environment. They are of three types: Human-Object Perception, Body-Motion Only, and Human-Robot Collaboration. The action sequences involved 11 subjects performing actions such as delivering the object to the robot, walking, picking up one of the two tools on the table, and interacting with the robot from a stationary position or moving in any direction.

2) *Description Model*: We use the SlowOnly neural network for action recognition, i.e., the Slow branch detached from SlowFast [4], with Resnet50 [18] as the backbone. It is a network pre-trained on Kinetics-400 [20], a dataset that

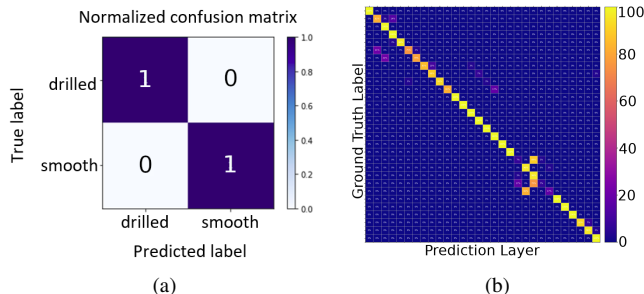


Fig. 5: Confusion matrices for the surface classification (a) and action recognition (b) models.

TABLE I: Comparison analysis on object surface dataset between baseline and proposed model.

Model	Resolution	Accuracy
Resnet-50	224	90
Resnet-152	224	94
Resnet-50 + SVM	224	<b>100</b>
Resnet-152 + SVM	224	<b>100</b>

TABLE II: Analysis comparing different approaches on the SlowOnly network for the HRI30 action recognition dataset.

Method	Backbone	Pretrained	Accuracy
SlowOnly	Resnet-50	Kinetics-400	<b>86.55</b>
SlowOnly	Resnet-50	Imagenet	64.29

consists of 240k training videos and 20k validation videos across 400 categories of human actions.

### C. 3D Human Body Tracking

To keep track of the human in the workspace, we focus on the 3D reconstruction of the human position. We extract the 3D position of human body joints by estimating it on the 2D position (u,v) transmitted from the Openpose neural network [21] via RGB-D camera. We simultaneously exploit each pixel's color and depth values of the input RGB-D data, keeping the data organized and filtering out from the point cloud the pixels lacking depth values (NaN) to address occlusion problems arising from the 3D mapping. Finally, we perform the extrapolation by setting the camera to a  $640 \times 480$  resolution with a frequency of 20Hz, having a one-to-one correspondence between depth and color data.

## V. EXPERIMENTS AND RESULTS

To validate the proposed framework, we employ the MOBILE Collaborative robotic Assistant (MOCA), a collaborative mobile manipulator ( $m = 6$  and  $n = 10$ ) composed of a Franka Emika Panda robotic arm and a Robotnik SUMMIT-XL STEEL mobile platform. Three Realsense d435i cameras operate at a frequency of 20Hz and a resolution of  $640 \times 480$ . In Sections V-A and V-B, we test the performances of the recognition systems. We explain the HRC task in Exp. 3, and in Exp. 4 we allow the human to control the robot by walking ergonomically in space. Finally, in Exp.5, we investigate the subjective workloads reported.

### A. Experiment 1: Surface Classification Model Benchmarking

In this experiment, we classify the surfaces of the objects handed by a human to a robot in the initial phase of the interaction. We start by splitting the dataset introduced in Sec. IV-A.3 into Train/Val/Test with a ratio of 70/20/10 and, as shown in Fig. 2.C, we preprocess all phases of the input images by using data augmentation techniques, such as scaling  $255 \times 255$  pixels, horizontal flipping, and center cropping crop  $224 \times 224$  pixels, to create virtual copies of different surface images and improve the generalization performance of the model. On the dataset, with a learning rate of 0.001 and an SGD (Stochastic Descending Gradient) optimizer, we train Resnet-50 and Resnet-152 to examine neural networks in both the baseline and proposed models. Firstly (Fig. 3), we freeze all the weights, i.e., knowledge gained in the pre-training on the ImageNet dataset, and cut the last three layers of the baseline ResNet-50/152, i.e., a fully connected (FC) layer of size 1000, a softmax layer and



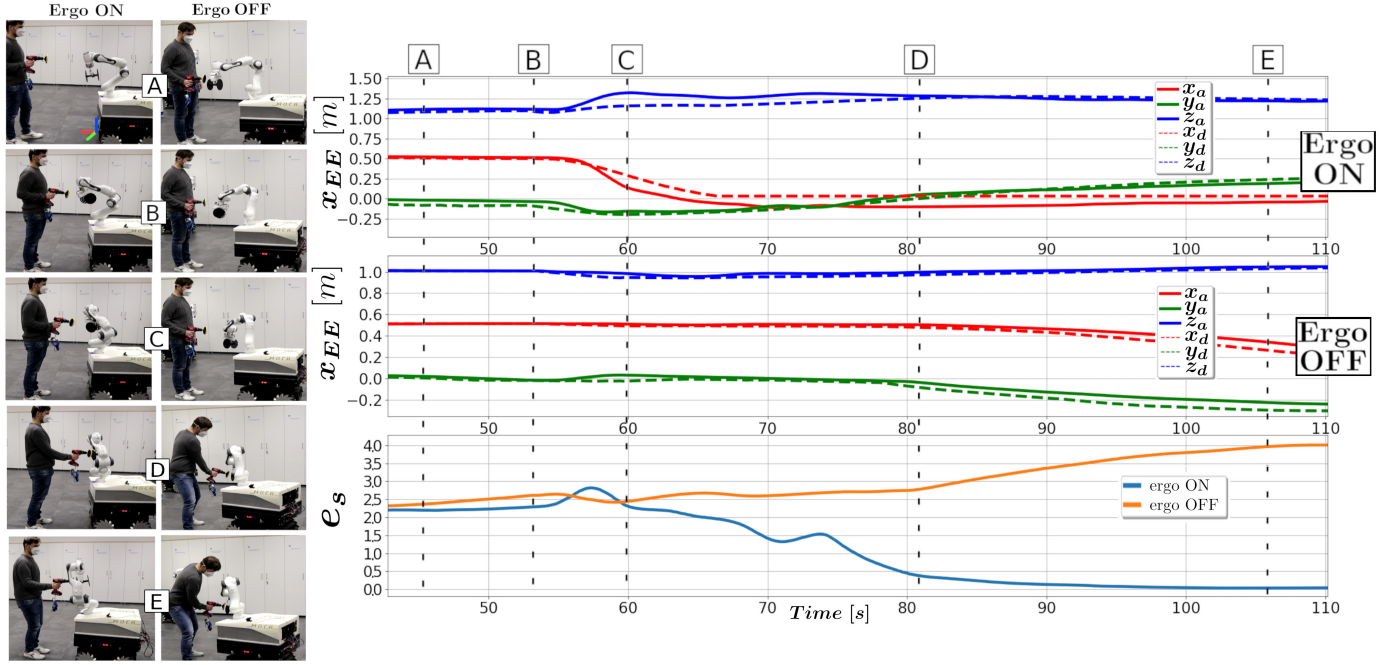


Fig. 6: Exp. 3. Object reorientation sequences, with (left column) and without (right column) optimal ergonomics (6). The three plots indicate respectively: EE trajectory for both cases with (top) and without (middle) optimal ergonomics, human ergonomics score (bottom).

the output layer. A new FC layer, a new softmax layer, and a new output layer are added. The FC layer of size two is connected to a dropout layer in the transferred ResNet.

In the second case, Fig. 4, the dataset is used to train the added SVM classifier by applying the same approach introduced in Sec. IV-A.4 on the ResNet-152 backbone. Hence, we connect the binary SVM classifier to the trained Average Pooling layer, trained using our dataset of object surface images on the previously described networks via transfer learning. We report the performance of the different implementations in Table I. The comparison shows that the baseline Resnet-152 outperforms the ResNet-50, given the number of layers that allow for more detailed feature extraction. Instead, the models based on our approach increased in accuracy by +6% for the ResNet-152 and by +10% for the Resnet-50 network, reaching 100% accuracy. A representation of this accuracy in class classification is shown in Fig. 5a. Despite the equality of accuracy between the backbones, we choose ResNet-50 as the backbone of our approach for its lower computational and model complexity.

### B. Experiment 2: Action Recognition Model Benchmarking

Here we analyze the performances of the neural network for action recognition. The dataset used to train the network was introduced in section IV-B.1. It includes 2940 action videos randomly divided into Train and Test with a ratio of 70 and 30. Before training and testing the network, the input frames are first sampled one every four frames per clip and then resized to  $256 \times 256$  pixels. Only for the training and tuning phases they are also randomly flipped. From these settings, we perform comparative performance analysis on different SlowOnly network models by applying a learning rate of 0.001 and a Stochastic Gradient Descent (SGD) optimizer [22] during network training. From the

comparison in Table II, we note that the network trained on the HRI30 dataset performs better when pretrained on kinetics-400 instead of ImageNet, and we justify this result by associating it to the different nature of the two pretrained datasets. While the former collects videos of human actions, the latter collects images of objects and animals. In addition, we use the confusion matrix in Fig. 5b to analyze the classification performance of each of the classes obtained by training the SlowOnly pre-trained network of Kinetics400 on the HRI30 dataset, from which it can be seen that the model achieved remarkable performance. Here, the diagonal cells represent the percentage of correctly classified test data with their true label, while the other cells represent the percentage of misclassified data. As shown, the model had difficulty in distinguishing between picking up and putting down the drill, object, or polisher, due to the similarity of actions. The datasets used in most industrial works are either not publicly available or different from those used in this work, so we cannot compare our results directly.

### C. Experiment 3: tasks order dictated by human intentions

In sequence, the subject has to first grab the object of Fig. 2 from a table and deliver it to the robot. The object has two surfaces, each one related to a specific operation to be performed on top (drilling for the perforated surface, polishing for the smooth surface). Thanks to the action recognition module, as soon as the subject delivers the object by straightening the arm (becoming collaborative), the robot grasps it. It then proceeds with surface identification through the onboard camera, which makes the robot aware of the orientation of the grabbed object, since this will affect the remainder of the experiment. Afterwards, based on the tool that the subject decides to use next (either drill or polisher), the robot will adjust the object's orientation, in

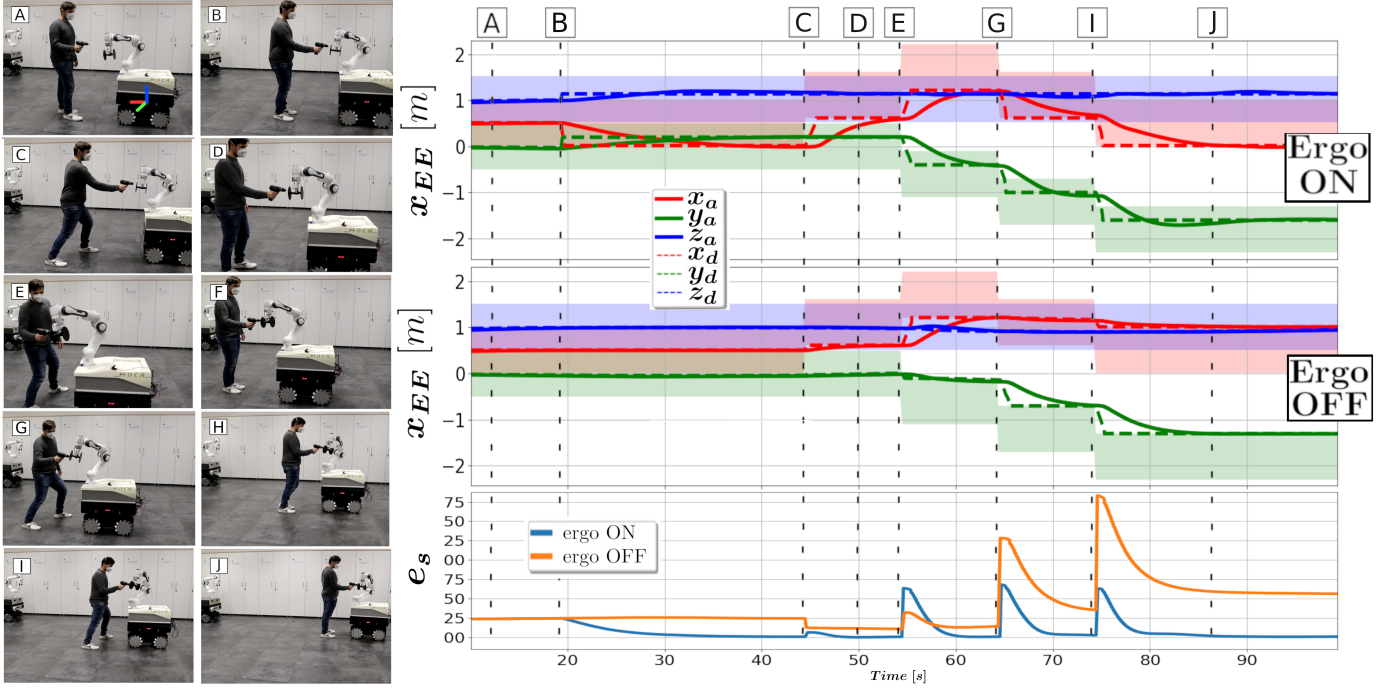


Fig. 7: Exp. 4. Snapshots sequence dictated by human intentions and actions. Plots: EE trajectory for both cases with (top) and without (middle) optimal ergonomics activation, human ergonomics score (bottom). The coloured regions represent the HRSW along each axis, based on the constraints on  $x_d$  defined in (8).

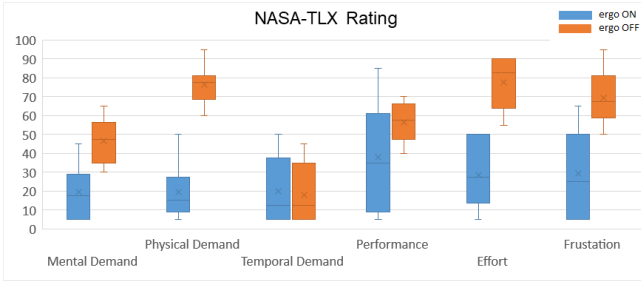


Fig. 8: NASA-TLX questionnaire outcomes on HRC task.

case the current surface facing the human is not matched with respect to the chosen tool (Fig. 6(A)). Since in the industrial environment, multiple people can carry the same tool, the robot will reorient the object only when the human becomes collaborative, meaning he/she directly engages in the interaction by straightening the arm (Fig. 6 (A)). Assuming no limiting occlusion, this increases the robustness when other people are in the frame.

If the tool held by the human does not match the current surface, the constraints region of the AHQP, in terms of orientation on  $x_d$ , will be modified through (7) in order to rotate the EE as in Fig. 6(C), while optimizing ergonomics. Given the limited range of motion of the robot's wrist and its initial position, a simple rotation of the last joint would be out of reach. Thereby, a full adjustment of the redundancy is necessary, which is correctly achieved by the AHQP.

To provide a benchmark, we perform the same experiment without the activation of optimal ergonomics (second column snapshots, Fig.6), thus only complying with the criteria of shortest distance from the actual pose, dictated by

$$\min_{\chi} \|\dot{x}_d\|^2, \quad (16)$$

which replaces (6). The bottom plot compares the ergonomics scores of the two cases, showing noticeable improvements, starting from the time instant (B) in which optimal ergonomics is activated. In (D) the human starts interacting, but shortly after (E), the score difference is already large, being  $e_s = 0.12$  for the proposed controller against  $e_s = 4.2$ , which indicates poor posture and high risk.

#### D. Experiment 4: Human-Following feature

Proceeding from Exp. 3, while the human is working on the object, he/she might decide to move in space (Fig. 7), e.g. to reach the next workstation, avoiding idle times. This puts the human in charge of the operation, accommodating movements and intentions during the task. Upon becoming collaborative (B), optimal ergonomics is activated, leading to a consistent reduction of  $e_s$  with respect to the benchmark. Thanks to the action recognition module, we can associate the subjects intentions to the movement of their feet. Indeed, during the HRC task, the user can decide to move around the robot by simply starting to walk in space (Fig. 7(C)). Once the action and its direction are detected, the constraints of the AHQP are changed by updating the HRSW towards the new intended direction in (8). This is seen in the  $x_{EE}$  plots of Fig. 7, in which at (C) the first step is detected (human backwards) and the constraints on  $x_d$  (8) are updated accordingly (robot forwards). The HRSW (position only) is coloured based on the  $x$ ,  $y$ , and  $z$  contribution, while the actual and desired EE trajectories are depicted with continuous and dashed lines respectively. Thanks to the hierarchical order defined in Sec. III-E, the EE remains in the feasible region, while optimizing ergonomics. The softened structure allows to exit from the HRSW during transitions, without infeasibility issues. It is also noticeable

how  $x_d$  behaves differently inside of the feasible region for the two cases, e.g., by looking at the  $y$ -component of  $x_d$ , we ensure not only the final solution to lie within the feasible region, but also to optimize ergonomics. The  $e_s$  plot, shows a cumulative increase of the score when ergonomics is inactive, clearly leading to harmful postures. The online control scheme proposed allows to continuously repeat this process (Fig. 1), allowing the robot to smoothly follow the subject. Hence, while optimally complying with the hierarchy, it is possible to command the robot by detecting the human body's inherent intentions.

#### E. Experiment 5: Usability evaluation

We here analyze the robustness of our approach across multiple subjects, to assess their perceived workload during Exp.s 3 and 4. This allows to quantify the benefits of the person-specific ergonomic maps, by selecting a proper variety of subjects. Indeed, ten healthy subjects (7 males and 3 females) conducted the experiments described in Sec.V-C and Sec.V-D. Their average age is 28.5 years, and their heights range from 1.58m to 1.94m. Without person-specific maps, such gap (0.36m) would clearly negatively impact the subjects experience, affecting the final scores. An initial training phase allowed the subjects to gain familiarity with the experiment, during which they expressed appreciation for the lack of uncomfortable wearable devices, that required initial calibration. Following the experiment, each subject was asked to fill out a NASA-TLX questionnaire form to assess how the experience was perceived. This is composed of the following scales scored from 0 to 20: temporal demand (TD), mental demand (MD), effort (EF), physical demand (PD), performance (PE), and frustration (FR). Fig. 8 shows improved overall PE (inverted scale), highlighting the ease of use of our framework in dynamic HRC. MD and PD are reduced as well, highlighting better execution responsiveness, and lower effort and stress levels. The data collection was carried out in accordance with the Declaration of Helsinki, and the protocol was approved by the ethics committee Azienda Sanitaria Locale (ASL) Genovese N.3 (Protocol IIT.HRI.ERGOLEAN 156/2020).

## VI. DISCUSSION AND CONCLUSIONS

The proposed framework identifies human actions and intentions through both surface classification and action recognition models. By formulating a hierarchical controller based on AHQP, we close the loop on the human figure, creating a framework that allows the operator to control the robot's redundancy by simply moving in space and letting the robot follow accordingly. The gathered data indicates noticeable ergonomics improvements, essential to reduce MSDs for future applications and further studies.

Possible limitations, besides common camera occlusion problems, involve the typical limitations of HQP when dynamically altering the stack of tasks. Indeed, optimal ergonomics must be carefully shifted in the hierarchy and the issue of continuity must be addressed in future studies.

## REFERENCES

- [1] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human-robot collaboration," *Autonomous Robots*, vol. 42, no. 5, pp. 957–975, 2018.
- [2] Q. Xiong, J. Zhang, P. Wang, D. Liu, and R. X. Gao, "Transferable two-stream convolutional neural network for human action recognition," *Journal of Manufacturing Systems*, vol. 56, pp. 605–614, 2020.
- [3] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, and J. Chen, "Deep learning-based human motion prediction considering context awareness for human-robot collaboration in manufacturing," *Procedia CIRP*, vol. 83, pp. 272–278, 2019.
- [4] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6202–6211.
- [5] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.
- [6] J. Zhang, H. Liu, Q. Chang, L. Wang, and R. X. Gao, "Recurrent neural network for motion trajectory prediction in human-robot collaborative assembly," *CIRP annals*, vol. 69, no. 1, pp. 9–12, 2020.
- [7] N. Mansard, O. Khatib, and A. Kheddar, "A unified approach to integrate unilateral constraints in the stack of tasks," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 670–685, 2009.
- [8] O. Khatib, L. Sentis, and J.-H. Park, "A unified framework for whole-body humanoid robot control with multiple constraints and contacts," in *European Robotics Symposium 2008*. Springer, 2008, pp. 303–312.
- [9] S. Gholami, F. Tassi, E. De Momi, and A. Ajoudani, "A reconfigurable interface for ergonomic and dynamic tele-locomanipulation," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 4260–4267.
- [10] H. Su, C. Yang, G. Ferrigno, and E. De Momi, "Improved human-robot collaborative control of redundant robot for teleoperated minimally invasive surgery," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1447–1453, 2019.
- [11] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.
- [12] O. Kanoun, F. Lamiroux, and P.-B. Wieber, "Kinematic control of redundant manipulators: Generalizing the task-priority framework to inequality task," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 785–792, 2011.
- [13] F. Tassi, E. De Momi, and A. Ajoudani, "Augmented hierarchical quadratic programming for adaptive compliance robot control," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3568–3574.
- [14] F. Tassi, E. De Momi, and A. Ajoudani, "An adaptive compliance hierarchical quadratic programming controller for ergonomic human-robot collaboration," *Robotics and Computer-Integrated Manufacturing*, 2022. [Online]. Available: [tinyurl.com/2p9y5jrz](https://tinyurl.com/2p9y5jrz)
- [15] B. Busch, M. Toussaint, and M. Lopes, "Planning ergonomic sequences of actions in human-robot interaction," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018.
- [16] S. Heydaryan, J. Souza Bedolla, and G. Belingardi, "Safety design and development of a human-robot collaboration assembly process in the automotive industry," *Applied Sciences*, vol. 8, no. 3, 2018.
- [17] W. Kim, M. Lorenzini, P. Balatti, Y. Wu, and A. Ajoudani, "Towards ergonomic control of collaborative effort in multi-human mobile-robot teams," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 3005–3011.
- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [19] F. Iodice, E. De Momi, and A. Ajoudani, "Hri30: An action recognition dataset for industrial human-robot interaction," 2022. [Online]. Available: [tinyurl.com/2p9y5jrz](https://tinyurl.com/2p9y5jrz)
- [20] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.
- [21] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.
- [22] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.