Dexterous Grasping by Manipulability Selection for Mobile Manipulator With Visual Guidance

Fei Chen, *Member, IEEE*, Mario Selvaggio, *Student Member, IEEE*, and Darwin G. Caldwell

***Abstract*—Industry 4.0 demands the heavy usage of robotic mobile manipulators with high autonomy and intelligence. The goal is to accomplish dexterous manipulation tasks without prior knowledge of the object status in unstructured environments. It is important for the mobile manipulator to recognize and detect the objects, determine manipulation pose, and adjust its pose in the workspace fast and accurately. In this research, we developed a stereo vision algorithm for the object pose estimation using point cloud data from multiple stereo vision systems. An improved iterative closest point algorithm method is developed for the pose estimation. With the pose input, algorithms and several criteria are studied for the robot to select and adjust its pose by maximizing its manipulability on a given manipulation task. The performance of each technical module and the complete robotic system is finally shown by the virtual robot in the simulator and real robot in experiments. This study demonstrates a setup of autonomous mobile manipulator for various flexible manufacturing and logistical scenarios.**

***Index Terms*—Manipulability, mobile manipulation, visual guidance.**

I. Introduction

MODERN Industry 4.0 manufacturing or Smart Manufacturing [2] demands the shift in manufacturing style from massive production to small quantity or massive customization [3]. In wage-intensive countries, this mode have created needs for automation and robotics with the feature of intelligence, dexterity, flexibility, and cost efficiency. The fact is that the current industrial robots are mainly deployed in factories doing repetitive and dirty works on a fixed point without much intelligence and flexibility. This introduced gap leads to the massive adoption of industrial level mobile manipulation robots with the capability of flexibility, mobility, and visual learning ability [4]. A robotic mobile manipulator is a robotic system built by one or two robotic arms fixed to a mobile platform. It allows performing tasks requiring locomotion and manipulation [5]. The new trend is to equip it with various sensors, e.g., laser, vision, tactile sensing to enhance its ability. Since 1980s, mobile manipulator development has experienced several stages with emphasis on different hardware or software components in different time. In the past years, various mobile manipulators have been developed. The most representative mobile manipulators are MORO [6], Rosie,[1] PR2,[2] Little Helper [7], KUKA OmniRob, and KMR iiwa[3] (see Fig. 1). For industrial application, KUKA OmniRob is selected as an experimental platform in many European flagship projects in robotics, such as First-MM,[4] TAPAS,[5] VALERI,[6] and EUROC,[7] targeting at high performance mobility and dexterous manipulation in the scenario of logistics and automation. It also demonstrates great potential for the application in disaster response, rescue, and in-home assistance. Driven by more recent challenging applications introduced by modern manufacturing style, e.g., logistical warehouse application introduced by the AMAZON picking challenge, multiple vision guided mobile manipulator shows promising performance. Various prototypes and products output from these activities reveal the fact that the modern mobile manipulation system is a highly integration of robotic technologies on perception [8], navigation [9], and learning and manipulation [10].

(a) (b)

Fig. 1. KUKA mobile manipulators as representative industrial mobile manipulators. (a) KUKA OmniRob, (b) KUKA KMR iiwa.

Mobile platforms have greatly extended the manipulator workspace and visual guidance provides plenty of clues for the robot to select proper pose to carry out a given grasping and

Manuscript received June 17, 2018; revised September 10, 2018; accepted October 15, 2018. Date of publication November 5, 2018; date of current version February 1, 2019. This work was supported by the European Union's Seventh Framework Programme under Grant agreement 608849 EuRoC. Paper no. TII-18-1577. *(Corresponding author: Fei Chen.)*

F. Chen and D. G. Caldwell are with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genova 16163, Italy (e-mail: fei.chen@iit.it; darwin.caldwell@iit.it).

M. Selvaggio is with the Department of Information Technology and Electrical Engineering, University of Naples Federico II, Napoli 80138, Italy (e-mail: mario.selvaggio@unina.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2018.2879426

[1] http://ias.cs.tum.edu/robots/tum-rosie/
[2] https://www.willowgarage.com/pages/pr2/
[3] https://www.kuka.com/
[4] http://www.first-mm.eu/
[5] http://www.tapas-project.eu/
[6] http://www.valeri-project.eu/
[7] http://www.euroc-project.eu/

manipulation tasks. Multimodal perception about the object and material is extremely important for these tasks.

Recent study solves the visual-tactile fusion problems for grasped object recognition using novel weakly paired sparse coding method [11]. A more general fusion framework was proposed in [12] to tackle the heterogeneous gap of visual, tactile, and auditory modalities for material identification. Such strategies are proved to be very effective for robust grasping and manipulation, however, it also increase the cost of the system to be adopted by industries.

For industrial mobile manipulator, the high redundancy in the system allows the robot to perform dexterous tasks using maximum manipulability. Since Yoshikawa introduced the manipulability measure in [13], this has been widely used in robotics for planning dexterous motions for robotic systems. Recent applications have shown a great interest in the integration of manipulability measure with flexible and self-collision capabilities. For instance, Vahrenkamp *et al.* in [14] integrated a planner to avoid collision for grasping motion by single and dual robotic arms. This planner integrates three main tasks necessarily required for grasping an object, which are as follows:

1) find a reasonable grasp pose;
2) resolve the inverse kinematics; and
3) search a collision-free trajectory.

An online grasp quality measure unit based on the computed applied forces, which diminishes the net torque, is used to score the quality of the grasping poses. More recently, manipulability analysis of humanoid robots has been carried out in [15]. The considered measure takes into account both joint limits and minimum distance among robot links and between links and obstacles. In this case, the manipulability is adopted to compute the quality distribution in the robot workspace considering redundancy and instantaneous direction of motion. In addition, in [16], an approach for inverting a precomputed representation of reachability so as to generate proper robot base poses for object grasping has been presented. Other works addressing the problem of optimal grasping pose selection can be found in the fields of robotic surgery [17], telemanipulation [18], and humanoid robots [19]. A crucial prerequisite for the successful execution of manipulation tasks in unstructured environments is the perception of the environment and the extraction of associated environmental affordances: in [20], an approach for the generation of whole-body manipulation actions based on the affordances extracted via multimodal exploration is presented. However, to the best knowledge of the authors, it is still seldom to see the study on modern visual guidance mobile manipulator for industrial applications.

II. Robot Description

A. Hardware Components

In this study, we use an industrial mobile manipulator, a new prototype produced by KUKA and German Aerospace Center (DLR) namely KUKA KMR iiwa, which is composed of a KUKA LBR iiwa collaborative arm, a KUKA KMR platform, and multiple stereo vision sensors [see Fig. 2(a)]. LBR iiwa is a new generation of lightweight robotic arm, a promising robotic arm for flexible industrial production process and safe, and compliant human and robot collaboration. It is particularly suitable for supporting human operator in industrial automation at ergonomically unfavorable workstations.

The KMR iiwa is equipped with two pairs of Mecanum wheels, so it can move in all directions, which has shortened robot throughput times and reduced nonproductive time when searching appropriate execution pose for a given task. This function opens up a wider range of choice for the entire new production mode and increases the cost effectiveness. With this technology, the KMR iiwa is able to move to a target position safely and smoothly with the speed of 4 m/s ahead or sideways and 2 m/s diagonally. It can still achieve the position accuracy with ± 1 mm even in constraint working space. The integrated laser range finder scans the surrounding environments, build the map, and embedded control software deals with the motion planning and navigation, which enables reliable and flexible work sequences. Therefore, it is significant that KMR iiwa mobile manipulator can utilize the efficiency and reliability of modern robotic technology for big industry automation.

The KMR iiwa is also equipped with two pairs of stereo vision modules to provide perception capability in both indoor and outdoor environments (see Fig. 2). One pair is mounted on the end-effector for detecting and recognizing the manipulated objects (namely eye-in-hand TCP stereo). The other pair together with a 2-DoF pan-tilt unit is built aside the robot arm to roughly localize the object and monitor the environment (namely eye-to-hand Pan-tilt stereo). Multiple stereo vision setup provides dense point cloud data and high quality color and depth image, and provides human eye completable ability for the robot to recognize the object even suffered by occlusion.

III. Manipulability With Visual Guidance

A. Pose Estimation With Stereo Vision

Large industries usually offer unfavorable and variant lighting conditions, thus lots of robotic assembly tasks are performed without visual guidance [21], [22]. In our application, it is mandatory for KMR iiwa to tolerate these challenges by developing robust industrial level solution by generating disparity matrices from stereo vision. The main principle is working like the human eyes. By computing the correspondences from two distinct views, left and right image frames, the relative relationship among features from both frames is exploited to extract the disparity image and further the depth image. Although disparity computation requires large amount of processing and might be influenced by various lighting conditions, its stable performance can always lead to acceptable results working in most of the industrial environment when compared with other sensor modules, which can also produce color and depth images [23].

Based on the RGB color image and depth image, point cloud data can be computed [24] as a input to the object pose estimation module. By improving the method introduced in [25] and [26], which computed the disparity image using pixelswise semiglobal matching, iterative closest points (ICP) algorithm is adopted to compute the pose of the object. In this paper, we use a small load carrier (SLC) as an example for testing

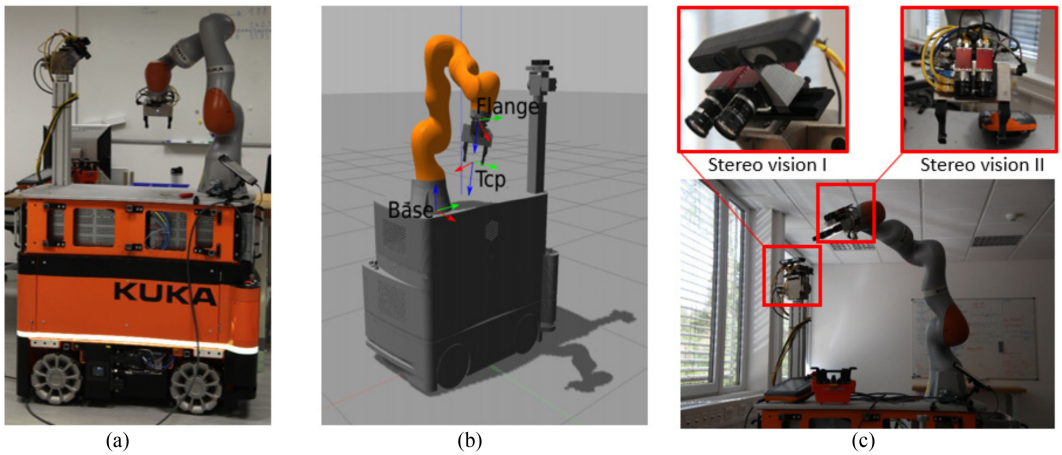

Fig. 2. Concept and setup of a mobile manipulator in this paper. (a) Industrial mobile manipulator. (b) Mobile robot in simulator. (c) Stereo vision setup.

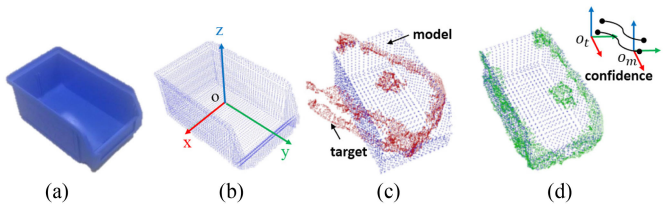

Fig. 3. Use ICP for pose estimation on point cloud data: an SLC example.

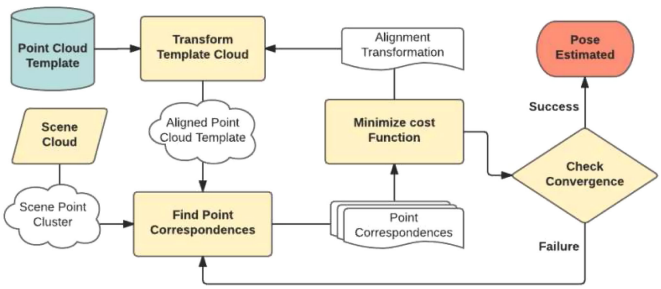

Fig. 4. Complete object recognition procedure using ICP.

the perception module. The concept of our proposed method on this object is shown in Fig. 3. The practical challenge of using ICP is that it easily leads to local minimum when searching for the optimal match of point cloud data samples if using real sensor data. Many interesting variations and extensions to this algorithm have been studied in the past years because the high modularity of this algorithm makes it possible to develop different approaches at each step of the pipeline, i.e., feature points selection, matching, and optimization [27] (see Fig. 4). We introduce a confidence metrics on computing the Euclidean distance between point cloud samples to evaluate the results. The definition of this confidence metric for a given registration is by computing the inverse of the average squared Euclidean distance between two groups of feature points

$$\text{Confidence} = \frac{n}{\sum_{i=0}^{n} \text{EuclideanDistance}(p_i^m, p_i^t)^2} \quad (1)$$

where n denotes the number of correspondences, p_i^m denotes the ith point of the group of feature points, and p_i^t denotes the ith point of the target group of feature points. With this confidence matrices, it allows us to easily compare the quality of different registrations to find out the best matching.

To avoid the local minimum of using ICP as a blind optimization, we further define an alignment space that is the set of admissible poses transformation of the reference point cloud frame to the target scene cloud frame. The idea is that once the identifying an optimal solution, a learning process by calculating the transformation between each local minima and the global one is triggered to study the error patterns in the alignment space. In order to correct this error, an error pattern transformation (EPT) term is defined as the transformation matrix transforming the wrong alignment into the correct one for each local minimum [28]. The advantage is that once EPT is computed, it can maintain valid regardless of the scene we are applying if we assume the target scene point cloud is always similar to the reference point cloud. Once final solution has been brought to the neighborhood of the optimal alignment, the EPT will be applied to the registration output and confidence metrics are then computed to evaluate all transformed poses. By selecting the one showing higher level of confidence, the correct transformation and thereby the correct pose is retrieved.

B. System Kinematics

For the user, it is always easy and intuitive to program a robot end-effector motion trajectory in the mobile manipulator operational space. A typical mobile manipulator, i.e., 7-DoF arm mounted on a 3-DoF mobile base, requires a whole body kinematic controller. Two main routines constitute the essential part of the whole body controller: the forward kinematic routine that gives the robot end-effector pose given the set of its generalized coordinates, and an inverse kinematics routine that solves for the group of joint angles for a given end-effector pose, usually expressed w.r.t. the robot base frame. When the robot is redundant, the latter calculation can be considered an optimization process in which the redundancy of the robot is exploited in

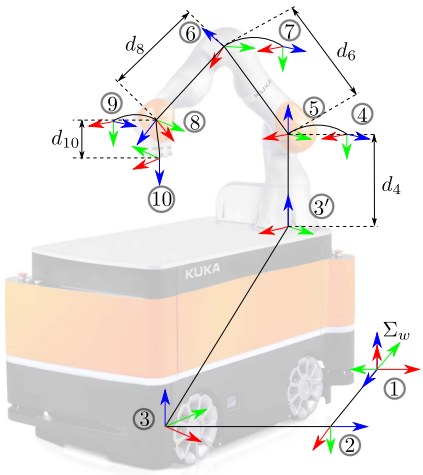

Fig. 5. KMR iiwa kinematic model representation.

TABLE I
KMR IIWA DH PARAMETERS (P = PRISMATIC JOINT; R = REVOLUTE JOINT; F = FIXED JOINT)

frame nr.	joint type	a	α	d	θ
1	P	0	$\pi/2$	–	$\pi/2$
2	P	0	$\pi/2$	–	$\pi/2$
3	R	0	$\pi/2$	0	–
3′	F	–	–	–	–
4	R	0	-$\pi/2$	0.36	–
5	R	0	$\pi/2$	0	–
6	R	0	-$\pi/2$	0.42	–
7	R	0	$\pi/2$	0	–
8	R	0	-$\pi/2$	0.4	–
9	R	0	$\pi/2$	0	–
10	R	0	-$\pi/2$	0.08	–

many ways to best adapt the robot motion behavior to any given task execution.

1) Forward Kinematics: The KMR iiwa forward kinematic model is derived by opportunely composing the kinematic models of both subsystems (robot arm and mobile platform).

The kinematic chain model of the KUKA KMRIIRA robot is shown in Fig. 5, whereas its Denavit–Hartemberg (DH) parameters are given in Table I.

By denoting the pose of the end-effector frame w.r.t. the arm base frame as $\boldsymbol{T}_{10}^{3'} \in \mathbb{R}^{4\times4}$ and the pose of the mobile platform frame with respect to the world frame as $\boldsymbol{T}_{3}^{w} \in \mathbb{R}^{4\times4}$, we can derive the whole body forward kinematic model as composition of homogeneous matrices, namely

$$\boldsymbol{T}_{10}^{w}(\boldsymbol{q}) = \boldsymbol{T}_{3}^{w}(\boldsymbol{q})\,\boldsymbol{T}_{3'}^{3}\boldsymbol{T}_{10}^{3'}(\boldsymbol{q}) \tag{2}$$

where all the $\boldsymbol{T}$'s are homogeneous transformation matrices, $\boldsymbol{T}_{3}^{3'} \in \mathbb{R}^{4\times4}$ is a fixed transformation that accounts for the pose of the arm mount point in the mobile base frame, that, in our case, can be written as follows:

$$T_{3'}^{3} = \begin{bmatrix} 0 & 1 & 0 & 0.16 \\ -1 & 0 & 0 & 0.37 \\ 0 & 0 & 1 & 0.965 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

and $\boldsymbol{q} \in \mathbb{R}^n$ is the vector of robot generalized coordinates (with n being the dimension of the robot joint space).

2) Inverse Kinematics: Let us denote with r the dimension of the robot task space. As stated previously, for redundant robots ($r < n$), the inverse kinematics can be computed by resorting to an optimization technique that solves for the set of generalized coordinates given an end-effector desired pose. The redundancy can be solved via a local optimization procedure of an opportunely defined objective function. For instance, when a reference joint velocity vector $\dot{\boldsymbol{q}}_0 \in \mathbb{R}^n$ is of interest, the problem can be formulated as an LQ problem, namely

$$\begin{aligned} \min_{\dot{\boldsymbol{q}}} \ & \tfrac{1}{2}\left(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_0\right)^T \boldsymbol{W}\left(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_0\right) \\ \text{s.t.} \ & \dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\,\dot{\boldsymbol{q}} \end{aligned} \tag{4}$$

where $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{r\times n}$ is the whole robot Jacobian, $\dot{\boldsymbol{q}} \in \mathbb{R}^n$ is the joint space velocities vector, $\dot{\boldsymbol{x}} \in \mathbb{R}^r$ is the operational space velocities vector, and $\boldsymbol{W} \in \mathbb{R}^{n\times n}$ is a weight matrix. This matrix can be opportunely defined to avoid joint limits [29] and self-collision [30]. If $\boldsymbol{W}$ is assumed to be the identity matrix (as in the rest of this paper) and $\dot{\boldsymbol{q}}_0 = \boldsymbol{0}$, the solution inherently minimize the Euclidean norm of the joint velocities $\dot{\boldsymbol{q}}$. The gradient projection method can be used to define $\dot{\boldsymbol{q}}_0$. We must recall that the solution to the problem in (4) can be written in the following form (for brevity we omit the dependence of the Jacobian upon the joint variables):

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{\dagger}\dot{\boldsymbol{x}} + \left(\boldsymbol{I} - \boldsymbol{J}^{\dagger}\boldsymbol{J}\right)\dot{\boldsymbol{q}}_0 \tag{5}$$

where $\boldsymbol{J}^{\dagger} = \boldsymbol{W}^{-1}\boldsymbol{J}^{\mathrm{T}}(\boldsymbol{J}\boldsymbol{W}^{-1}\boldsymbol{J}^{\mathrm{T}})^{-1}$ is the Jacobian weighed right pseudoinverse. The first term in the sum in (5) represents the least velocity norm solution and the second the orthogonal projection of a possible secondary task $\dot{\boldsymbol{q}}_0$ into the null-space of $\boldsymbol{J}$. The projection of $\dot{\boldsymbol{q}}_0$ onto the null-space of $\boldsymbol{J}$ ensures that the secondary (lower priority) task will not affect the primary (higher priority) task (e.g., satisfy the end-effector pose constraint). The choice of $\dot{\boldsymbol{q}}_0$ can be derived from differentiation of an additional objective function $H(\boldsymbol{q})$ which is tried to be maximized/minimized by the inverse kinematic algorithm. Most typical objective functions may come from following criteria:

1) manipulability → maximize distance from singularities;
2) joint range → maximize distance from joint limits;
3) obstacle avoidance → maximize distance from objects;
4) given configuration → minimize distance from a given configuration.

In this framework, the task priority plays a significant role. Using this formulation, we address the highest priority task and try to satisfy secondary tasks. Addition, deletion, or reordering of the tasks would be possible and convenient to cope with multiple scenarios [31].

C. Manipulability Criterion

Manipulability index (from now on indicated by the symbol w), first introduced by Yoshikawa in [13], represents a value that embeds information about the capability of the robot end-effector to change its position and orientation unconditionally. This information is inherently contained into the Jacobian matrix $\boldsymbol{J}$ and, in particular, in its singular values $\boldsymbol{s} = [s_1, s_2, \ldots, s_n]$,

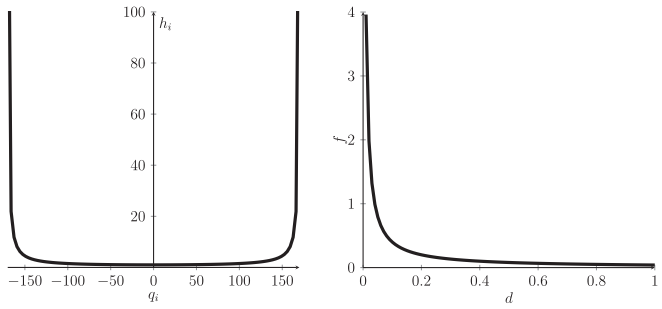

Fig. 6. Plot of the performance criteria used for joint limits (left-hand side) and minimum distance (right-hand side). In this example, $q_{i,\max} = 170°$, $q_{i,\min} = -170°$, $\gamma = 4$, $\rho = 1/25$, $\alpha = 0$, $\beta = 25$.

namely

$$w = \sqrt{\det(\boldsymbol{J}\boldsymbol{J}^T)} = s_1 s_2 \dots s_n. \tag{6}$$

When used as a criterion to solve the inverse kinematic problem, through the gradient projection method, manipulability can serve as an additional objective function, to be maximized by the solution. In this way, the redundancy of the robot can be effectively exploited to favor joint space configurations that correspond to higher manipulability index. However, manipulability criterion can be used to support a decision process, such as grasp selection in manipulation tasks. The manipulability measure can be efficiently combined with the reachability information that can be used by the robot to select a reachable end-effector pose that maximizes the manipulability of a picking task [15]. The computation of these useful metrics can be opportunely carried out in an offline process that stores the computed information in a database within a voxelized three-dimensional (3-D) or 6-D data structure representing the robot workspace. In addition, an extended manipulability measure can be derived that accounts for joint limitations and self-collisions following the approach proposed in [15]. More specifically, by defining appropriate penalization functions, the Jacobian matrix entries can be accordingly penalized and so is the manipulability index. The functions used in this paper have been derived from [30] and are resumed in the following sections.

1) Joint Limits: A cost function $h(\boldsymbol{q}) : \mathbb{R}^n \to \mathbb{R}$ that models joint limits nearness must tend to infinity as one of the joint variables approach its limits. Among many possibilities, we adopt the following function:

$$h(\boldsymbol{q}) = \sum_{i=1}^{n} h_i(q_i) = \sum_{i=1}^{n} \frac{1}{\gamma} \frac{(q_{i,\max} - q_{i,\min})^2}{(q_{i,\max} - q_i)(q_i - q_{i,\min})} \tag{7}$$

where n is the number of joints, γ is a scalar gain, q_i is the ith joint coordinate, $q_{i,\max}$ and $q_{i,\min}$ are the upper and the lower limits, respectively (see Fig. 6, left-hand side). Its gradient $\nabla h(\mathbf{q})$ represents the direction of fastest increase of $h(\mathbf{q})$. The ith component of the gradient can be written as

$$\frac{\partial h(\boldsymbol{q})}{\partial q_i} = \frac{1}{\gamma} \frac{(q_{i,\max} - q_{i,\min})^2 (2q_i - q_{i,\max} - q_{i,\min})}{(q_{i,\max} - q_i)^2 (q_i - q_{i,\min})^2}. \tag{8}$$

It can easily be noted that $\partial h(\boldsymbol{q})/\partial q_i$ attains zero at the middle of the ith joint and goes to infinity to either limits.

2) Self-Collision: Similarly to the joint limits case, we seek for a self-collision cost function $f(d(\boldsymbol{q})) \in \mathbb{R}^n \to \mathbb{R}$ that inherently depends upon the minimum distance among all the robot links $d(\geq 0)$ and tends to infinity as it approaches the value $d = 0$. The gradient of f can be calculated as follows (we omit the dependencies on the right-hand side of the equations):

$$\nabla f(d(\boldsymbol{q})) = \frac{\partial f}{\partial \boldsymbol{q}} = \frac{\partial f}{\partial d} \frac{\partial d}{\partial \boldsymbol{q}}. \tag{9}$$

In case of self-collision, the second term in (9) can be further specified in

$$\frac{\partial d(\boldsymbol{q})}{\partial \boldsymbol{q}} = \frac{1}{d}[\boldsymbol{J}_A^T(\boldsymbol{p}_A - \boldsymbol{p}_B) + \boldsymbol{J}_B^T(\boldsymbol{p}_B - \boldsymbol{p}_A)]^{\mathrm{T}} \tag{10}$$

where $\boldsymbol{p}_A \in \mathbb{R}^3$ and $\boldsymbol{p}_B \in \mathbb{R}^3$ are the position vectors of the two nearest points on the robot links expressed in the base frame (obtained using a collision detection and minimum distance calculation routine), $\boldsymbol{J}_A \in \mathbb{R}^{3\times n}$ and $\boldsymbol{J}_B \in \mathbb{R}^{3\times n}$ are the Jacobian matrices associated with the points $\boldsymbol{p}_A$ and $\boldsymbol{p}_B$, respectively. In this paper, we use the Gilbert–Johnson–Keerthi algorithm for proximity queries [32], [33]. A possible choice for the $f(d(\boldsymbol{q}))$ is given by

$$f(d(\boldsymbol{q})) = \rho e^{-\alpha d} d^{-\beta} \tag{11}$$

where α, β, and ρ are parameters that control the function trend (see Fig. 6, right-hand side). The partial derivative of this function with respect to d is

$$\frac{\partial f(d(\boldsymbol{q}))}{\partial d} = -\rho e^{-\alpha d} d^{-\beta}(\beta d^{-1} + \alpha). \tag{12}$$

At this point, $\nabla f(d(\boldsymbol{q}))$ can be computed using (9), (10), and (12).

3) Extended Manipulability: In order to build a manipulability measure that incorporate information about joint limits and self-collision, weight matrices have to be calculated. These matrices will be useful to build the augmented Jacobian $\tilde{\boldsymbol{J}}$ that gives the extended manipulability measure. Since, this quantity contains information influenced by the chosen end-effector motion direction, it inherently depends on the specified movement. As in [15], our purpose is to have a representation that covers all the possible movements. To this end, the workspace movements can be partitioned by 2^6 hyperoctants identified by $\Gamma \in \{-1, 1\}$. For each Γ, an augmented Jacobian can be built and the extended manipulability, consequently, extracted.

We now present the derivation of the two penalization terms that are essential for the augmented Jacobian computation. The joint penalization matrix $\boldsymbol{L}(\Gamma, \boldsymbol{q})$ accounts for the presence of joint limits and can be represented as follows:

$$L_{i,j} = \begin{cases} p_j^-, & \text{sign}(J_{i,j}(\boldsymbol{q}))\text{sign}(\Gamma_i) < 0 \\ p_j^+, & \text{otherwise} \end{cases} \tag{13}$$

where

$$p_j^- = \begin{cases} 1, & |q_j - q_{j,\min}| > |q_{j,\max} - q_j| \\ \frac{1}{\sqrt{1+|\nabla h(q)_j|}}, & \text{otherwise} \end{cases}$$
$$p_j^+ = \begin{cases} \frac{1}{\sqrt{1+|\nabla h(q)_j|}}, & |q_j - q_{j,\min}| > |q_{j,\max} - q_j| \\ 1, & \text{otherwise.} \end{cases} \quad (14)$$

These formulas express the fact that, having chosen an end-effector motion direction, if the considered joint is moving away from its limits, the associated penalization term $L_{i,j} = 1$ (no penalization), if it is moving toward one of its limits, the penalization is applied $L_{i,j} < 1$. On the other hand, an obstacle penalization matrix $\boldsymbol{O}(\Gamma, \mathbf{q})$ accounts for the presence of obstacles and can be computed as follows:

$$O_{i,j} = \begin{cases} o_j^-, & \text{sign}(\Gamma_i) < 0 \\ o_j^+, & \text{otherwise} \end{cases} \quad (15)$$

where

$$o_j^- = \begin{cases} 1, & v_i > 0, i > 3 \\ \frac{1}{\sqrt{1+|\nabla f(q)_j|}}, & \text{otherwise} \end{cases}$$
$$o_j^+ = \begin{cases} \frac{1}{\sqrt{1+|\nabla f(q)_j|}}, & v_i > 0, i \leq 3 \\ 1, & \text{otherwise} \end{cases} \quad (16)$$

where v_i is the ith component of the velocity vector and $\boldsymbol{v} = \dot{\boldsymbol{p}}_A - \dot{\boldsymbol{p}}_B$ is the relative linear velocity of nearest points $(\boldsymbol{p}_A, \boldsymbol{p}_B)$. Also in this case, we can give an intuitive explanation of the above-mentioned formulas. If the chosen end-effector motion direction makes the minimum distance increase, the corresponding weight is $O_{i,j} = 1$, otherwise penalization is applied and $O_{i,j} < 1$.

Finally, the augmented Jacobian can be computed as

$$\tilde{J}_{i,j}(\Gamma, \boldsymbol{q}) = L_{i,j}(\Gamma, \boldsymbol{q}) O_{i,j}(\Gamma, \boldsymbol{q}) J_{i,j}(\boldsymbol{q}). \quad (17)$$

The considered Jacobian $\tilde{\boldsymbol{J}}(\Gamma, \boldsymbol{q})$ is, therefore, decomposed via singular value decomposition in order to retrieve the corresponding set of singular values $\tilde{\boldsymbol{s}}$ and the extended manipulability measure can be calculated as follows:

$$\tilde{w} = \sqrt{\det(\tilde{\boldsymbol{J}}\tilde{\boldsymbol{J}}^T)} = \tilde{s}_1 \tilde{s}_2 \ldots \tilde{s}_n. \quad (18)$$

4) Simulation: Some simulation experiments are carried out to show the capability of the methods and highlight the main features using the KMR iiwa mobile manipulator (see Section II-A). We calculate the manipulability and extended manipulability measures by sampling the arm robot workspace. This calculation is useful when robots have to operate in the environment and their online grasp selection capabilities can be enhanced consulting a map of offline generated manipulability measures. The results of these experiments are shown in Fig. 7 and aim to give to the reader some more insightful intuition about different manipulability measures. For each experiment, 50 000 cells in the robot workspace have been considered. These cells correspond to the same number of randomly generated joint values inside the joint limitations range. In Fig. 7(a), the classical manipulability measure is shown by means of colored points. Fig. 7(b) contains the manipulability penalized solely through the joint limit penalization matrix. In this experiment, conventional KMR iiwa arm joint limits have been considered. They are shown in Table II.

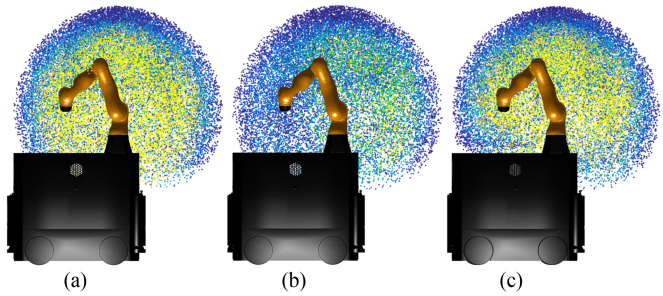

Fig. 7. Manipulability measure the graphical representation for the robot arm calculated for three different cases. In (a), the classical manipulability, in (b), the penalization terms due to joint limits has been applied, and in (c), minimum distance penalization is taken into account. Yellow colored points represent higher manipulability index configurations.

TABLE II
KUKA IIWA ARM JOINT LIMITS

Joint	1	2	3	4	5	6	7
Limits	±170°	±120°	±170°	±120°	±170°	±120°	±170°

Hyperoctant $\Gamma = [1, 1, 1, 1, 1, 1]$ is considered in this experiment. As it can be noticed, the manipulability significantly decreases due to joint limits penalizations. The last experiment is shown in Fig. 7(c) where self-collision is also considered. To have a clearer representation, we considered the simplified case in which only the end-effector and mobile platform minimum distance is considered. This is also motivated by the fact that end-effector collisions are more likely to happen with mobile base than with other links. The value used for (11) are as follows: $\alpha = 1$; $\beta = 1$; and $\rho = 0.2$. The exploited hyperoctant is $\Gamma = [-1, 1, -1, 1, 1, 1]$. From the considered figure, it can be noticed that the penalization term implies a manipulability reduction when the end-effector is closer to the mobile base top part.

IV. EXPERIMENTS

A. Manipulability Based Pose Selection

The experiment presented in this section assume a grasping/manipulation task of an SLC from a shelf. The pose estimation module is able to give the target object pose with respect to the robot base frame. Even though more complicated solution can be exploited for grasping and manipulation of complex objects, the geometrical simplicity of the SLC allows the use of an heuristic approach to both select grasp points in the SLC reference frame and define pre- and postgrasp movements representing main manipulation tasks. The problem we aim to solve is to dexterously accomplish the grasping/manipulation task using such a high redundant robot with vision guidance. The problem of high redundancy introduced by the used robot is decoupled, i.e., the arm redundancy is always solved in a

Algorithm 1: Mobile Platform Placement Computation.

Require: *samplePoseEE*, *areaWidth*, *areaHeight*
  platformPoses ← sample(*areaWidth,areaHeight*)
  for all posMP **in** platformPoses **do**
    moveVirtualPlatform(posMP)
    reachable, joints ←
    inverseKinematics(*samplePoseEE*)
    if reachable **then**
      manip ← manipulability(*joints*)
    end if
  end for
  index ← max(manip)
  return moveRealPlatform(platformPoses(index))

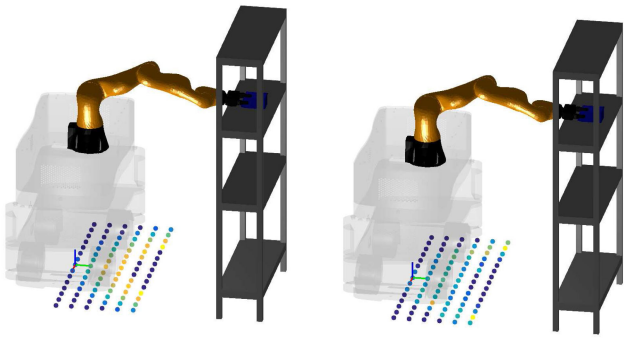

Fig. 8. (a) Manipulability and (b) extended manipulability measure computed for different mobile platform poses during the preprocessing of the task. Blue dots indicate the poses in which the object is not reachable and yellow dots represent the poses with higher manipulability index.

successive step with respect to the mobile base platform placement. Hence, the first choice to take in consideration is the placement of the mobile base in the shelf manipulation area. The working area bounds can be optimally defined *a priori* on the base of geometrical considerations and/or based on reachability information. We discretized the floor area surrounding the shelf where the target SLC is placed, and compute the inverse kinematics solution that maximizes the arm extended manipulability for each mobile base placement. The maximum manipulability of the arm is returned and stored to be used then as metric to evaluate the mobile platform placement. This allows the robot to build a map containing two important information in each cell: reachability (1 corresponding to the existence of an inverse kinematics solution, 0 otherwise) and manipulability (scalar real numbers).[8] The overall procedure for the platform placement is summarized in Algorithm 1. Fig. 8 contains the results of the proposed experiment in which both manipulability and extended manipulability are shown for each mobile platform pose. For the sake of clarity of the presented results, we show the experiment in which we keep the orientation of the mobile platform fixed

[8]It is worth noting that the null reachability does not implies null manipulability, since the inverse kinematics can still return a configuration which tends toward the grasp pose. Thus, it is necessity to consider first reachable poses and then choose the one that maximizes the manipulability measure.

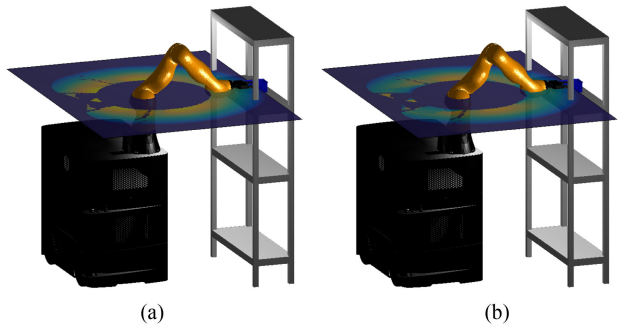

Fig. 9. (a) Manipulability and (b) extended manipulability measure computed in the plane containing grasping poses. Blue zone indicates the poses in which the object is not reachable and yellow zones represent the end-effector poses with higher manipulability index.

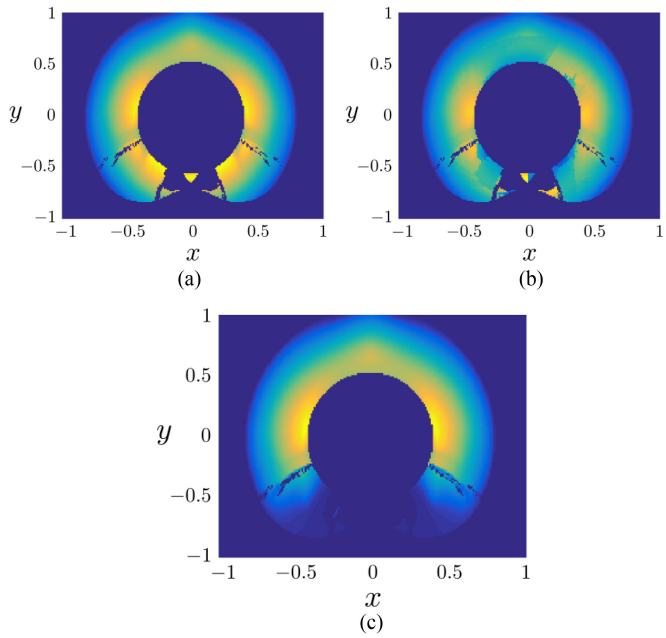

Fig. 10. (a) Manipulability and (b) extended manipulability measure considering self-collision and (c) joint limits computed in the plane containing grasping poses. Blue zone indicates the poses in which the object is not reachable and yellow zones represent the end-effector poses with higher manipulability index.

letting only the candidate positions to change. Further experiments with orientation variation have been performed and the results do not deviate significantly from the proposed ones.

To better show the influence of self-collision and joint limits onto the extended manipulability measure, we computed the manipulability of the arm moving the end-effector in the plane containing the grasping poses (see Figs. 9 and 10). In this experiment, we have used the criterion of minimizing the distance from an upper elbow configuration. We used a reference joint position vector $\boldsymbol{q}_A = [0, 45, 0, 90, 0, -45, 0]$ degree (corresponding to upper elbow configuration) and compute the inverse kinematic map using the projected gradient control in (5) with $\dot{\boldsymbol{q}}_0 = \nabla H(\boldsymbol{q})$ and $H(\boldsymbol{q}) = (\boldsymbol{q}_A - \boldsymbol{q})^{\mathrm{T}} \boldsymbol{K}_A (\boldsymbol{q}_A - \boldsymbol{q})$ with $\boldsymbol{K}_A$ being a diagonal positive definite gain matrix.

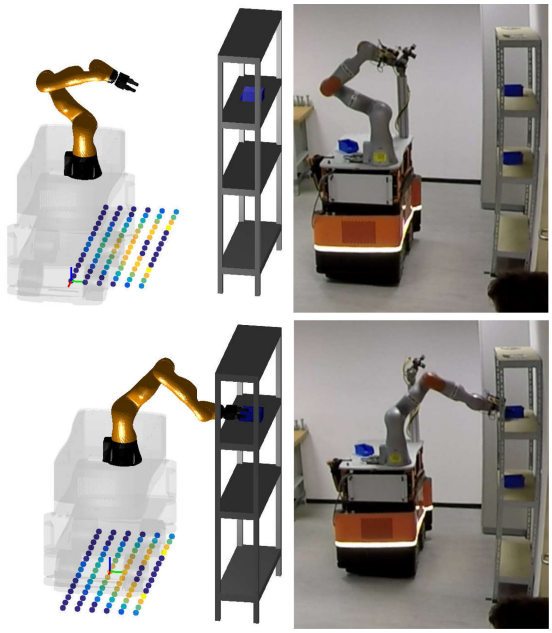

Fig. 11. Results of the experiments with manipulability based platform pose selection for object grasping. The left-hand side column shows the mobile robot and its manipulability measure during the grasp execution (right-hand side column).

B. Discussion

Manipulability and the extended manipulability measures have been used to perform simulated and real case experiments. In both cases, the platform placement process considers 90 candidate grasping poses in which manipulability measure and reachability information have been exploited. Fig. 8 shows the results of simulated experiments as colored dots on the floor: blue dots indicates platform poses in which the object is not reachable, whereas yellow dots represent the poses with high manipulability index. The grasp process encloses a mobile platform placement computation, which is reported in Algorithm 1. More precisely, once the pose of the object to be grasped is estimated, the computation starts and gives the best mobile platform placement for the grasping/manipulation task. However, it can be noticed that, maximum manipulability can be achieved for mobile platform poses that are very close to the shelf. In the real case, for safety reasons, it may be reasonable to maintain a certain distance from the shelf. This has been achieved by choosing a platform pose that is optimal in terms of manipulability and minimize the distance from the starting platform pose. This is also motivated by the fact that, assuming the platform is already in front of the shelf (see reference frame in Fig. 8), less distance has to be covered speeding-up the task accomplishment. Results of the real experiments regarding the optimal platform pose selection are shown in Fig. 11.

On the other hand, Fig. 9 gives a visual representation about the step arm manipulability. More detailed graphs are shown in Fig. 10 where a comparison among different manipulability measures is given. In this case, only the considered hyperoctant has been changed with respect to the previous sections to take into account the movements in the direction orthogonal to the shelf (movements needed for pregrasp and postgrasp phases).

V. Conclusion

In this paper, we present a modified and improved error pattern transformation based on iterative closest point algorithm to estimate the targeted object pose on the point cloud data retrieved from multiple stereo vision systems. The algorithm can always find out the correct object pose when examining an SLC grasping example by KMR iiwa robot. Moreover, given the grasping pose that is associated with the object pose, robot pose is selected to maximize the manipulability of the end-effector with proposed manipulability modeling methodologies. Experiments show the convincing performance of each proposed technical method. Future work will focus on the improvement of the controller for online manipulability computation and selection.

References

[1] B. Siciliano *et al.*, "Euroc-the challenge initiative for european robotics," in *Proc. 41st Int. Symp. Robot.*, 2014, pp. 1–7.

[2] K. Schwab, *The Fourth Industrial Revolution* World Econ. Forum, *Geneva*, Switzerland, 2016.

[3] A. G. C. Gonzalez, M. V. S. Alves, G. S. Viana, L. K. Carvalho, and J. C. Basilio, "Supervisory control-based navigation architecture: A new framework for autonomous robots in industry 4.0 environments," *IEEE Trans Ind. Informat.*, vol. 14, no. 4, pp. 1732–1743, Apr. 2018.

[4] M. Hvilshøj, S. Bøgh, O. Skov Nielsen, and O. Madsen, "Autonomous industrial mobile manipulation (AIMM): Past, present and future," *Ind. Robot, Int. J.*, vol. 39, no. 2, pp. 120–135, 2012.

[5] F. Ficuciello, R. Carloni, L. C. Visser, and S. Stramigioli, "Port-Hamiltonian modeling for soft-finger manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2010, pp. 4281–4286.

[6] J. Schuler, *Integration Von Förder-und Handhabungseinrichtungen*, vol. 104. Berlin, Germany: Springer-Verlag, 2013.

[7] M. Hvilshøj, S. Bøgh, O. Madsen, and M. Kristiansen, "The mobile robot little helper: Concepts, ideas and working principles," in *Proc. IEEE Conf. Emerg. Technol. Factory Automat.*, 2009, pp. 1–4.

[8] T. N. Shene, K. Sridharan, and N. Sudha, "Real-time surf-based video stabilization system for an FPGA-driven mobile robot," *IEEE Trans. Ind. Electron.*, vol. 63, no. 8, pp. 5012–5021, Aug. 2016.

[9] Y. Toda and N. Kubota, "Self-localization based on multiresolution map for remote control of multiple mobile robots," *IEEE Trans Ind. Informat.*, vol. 9, no. 3, pp. 1772–1781, Aug. 2013.

[10] C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang, "A learning framework of adaptive manipulative skills from human to robot," *IEEE Trans Ind. Informat.*, 2018, to be published, [Online]. Available: https://ieeexplore.ieee.org/document/8336935

[11] H. Liu, Y. Yu, F. Sun, and J. Gu, "Visual–tactile fusion for object recognition," *IEEE Trans. Automat. Sci. Eng.*, vol. 14, no. 2, pp. 996–1008, Apr. 2017.

[12] H. Liu, F. Sun, B. Fang, and S. Lu, "Multi-modal measurements fusion for surface material categorization," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 2, pp. 246–256, Feb. 2018.

[13] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. Robot. Res.*, vol. 4, no. 2, pp. 3–9, 1985.

[14] N. Vahrenkamp, M. Do, T. Asfour, and R. Dillmann, "Integrated grasp and motion planning," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2010, pp. 2883–2888.

[15] N. Vahrenkamp, T. Asfour, G. Metta, G. Sandini, and R. Dillmann, "Manipulability analysis," in *Proc. 12th IEEE-RAS Int. Conf. Humanoid Robots*, Nov. 2012, pp. 568–573.

[16] N. Vahrenkamp, T. Asfour, and R. Dillmann, "Robot placement based on reachability inversion," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2013, pp. 1970–1975.

[17] T. Liu and M. C. Avuolu, "Optimal needle grasp selection for automatic execution of suturing tasks in robotic minimally invasive surgery," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2015, pp. 2894–2900.

[18] M. Selvaggio, G. Notomista, F. Chen, B. Gao, F. Trapani, and D. Caldwell, "Enhancing bilateral teleoperation using camera-based online virtual fixtures generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2016, pp. 1483–1488.

[19] F. Burget and M. Bennewitz, "Stance selection for humanoid grasping tasks by inverse reachability maps," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2015, pp. 5669–5674.
[20] P. Kaiser, D. Gonzalez-Aguirre, F. Schltje, J. Borrs, N. Vahrenkamp, and T. Asfour, "Extracting whole-body affordances from multimodal exploration," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 1036–1043.
[21] J. Huang, Y. Wang, and T. Fukuda, "Set-membership-based fault detection and isolation for robotic assembly of electrical connectors," *IEEE Trans. Automat. Sci. Eng.*, vol. 15, no. 1, pp. 160–171, Jan. 2018.
[22] F. Chen, L. Carbonari, C. Canali, M. D'Imperio, and F. Cannella, "Design of a novel dexterous robotic gripper for in-hand twisting and positioning within assembly automation," *Assembly Automat.*, vol. 35, no. 3, pp. 259–268, 2015.
[23] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald, "Review of stereo vision algorithms and their suitability for resource-limited systems," *J. Real-Time Image Process.*, vol. 11, no. 1, pp. 5–25, 2016.
[24] Z. Jakovljevic, R. Puzovic, and M. Pajic, "Recognition of planar segments in point cloud based on wavelet transform," *IEEE Trans Ind. Informat.*, vol. 11, no. 2, pp. 342–352, Apr. 2015.
[25] H. Hirschmüller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 2, pp. 807–814.
[26] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Auton. Robots*, vol. 34, no. 3, pp. 133–148, 2013.
[27] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proc. 3rd Int. Conf. 3-D Digit. Imag. Model.*, 2001, pp. 145–152.
[28] B. Gao, F. Chen, F. Trapani, M. Selvaggio, and D. Caldwell, "Robust object localization based on error patterns learning for dexterous mobile manipulation," in *Proc. Int. Conf. Adv. Robot. Mechatronics*, 2016, pp. 213–218.
[29] T. F. Chan and R. V. Dubey, "A weighted least-norm solution based scheme for avoiding joint limits for redundant manipulators," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 1993, vol. 3, pp. 395–402.
[30] B. Dariush, G. B. Hammam, and D. Orin, "Constrained resolved acceleration control for humanoids," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 710–717.
[31] P. Chiacchio, S. Chiaverini, L. Sciavicco, and B. Siciliano, "Closed-loop inverse kinematics schemes for constrained redundant manipulators with task space augmentation and task priority strategy," *Int. J. Robot. Res.*, vol. 10, no. 4, pp. 410–425, 1991.
[32] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, "A fast procedure for computing the distance between complex objects in three-dimensional space," *IEEE J. Robot. Automat.*, vol. 4, no. 2, pp. 193–203, Apr 1988.
[33] E. G. Gilbert and C. P. Foo, "Computing the distance between general convex objects in three-dimensional space," *IEEE Trans. Robot. Automat.*, vol. 6, no. 1, pp. 53–61, Feb. 1990.

Fei Chen (S'09–M'12) received the B.S. degree in computer science from Xian Jiaotong University, Xian, China, in 2006, the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2008, and the Dr. Eng. degree in robotics from Fukuda Laboratory, Department of Micro-Nano Systems Engineering, Nagoya University, Nagoya, Japan, in 2012, followed by an one year post-doc.

In June 2013, he joined the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Italy. Since then, he has been leading the Mobile Manipulation Group focusing on applied robot perception and learning for robotic mobile manipulators. He is the PI of AutoMAP project which is funded by EU FP7 EUROC project. His main research interests include robot manipulation, robot vision, robot learning, and human–robot collaboration.

Mario Selvaggio (S'16) received the Bachelor's and the Master's degrees (*magna cum laude*) in mechanical engineering, in 2013 and 2015, respectively, from the University of Naples Federico II, Naples, Italy, where he is currently working toward the Ph.D. degree in information technology and electrical engineering.

In 2014, he was a master thesis student with Fraunhofer IGD, Darmstadt, Germany, and in 2016, he was an intern with the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genova, Italy. His research activities focus on shared control, robotic teleoperation, port-Hamiltonian modeling, passivity-based control, and grasping and manipulation.

Darwin G. Caldwell received the B.S. and Ph.D. degrees in robotics from the University of Hull, Hull, U.K., in 1986 and 1990, respectively, and the M.Sc. degree in management from the University of Salford, Salford, U.K., in 1996.

He is currently the Director of the Department of Advanced Robotics, Istituto Italiano di Tecnologia, Genova, Italy. He is a Visiting/Honorary/Emeritus Professor with the University of Sheffield, the University of Manchester, and the University of Wales, Bangor. His research interests include innovative actuators and sensors, haptic feedback, force augmentation exoskeletons, dexterous manipulators, humanoid robotics, biomimetic systems, rehabilitation robotics, telepresence and teleoperation procedures, and robotics and automation systems for the food industry.