ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Integration of robotic vision and tactile sensing for wire-terminal insertion tasks

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:
De Gregorio, D., Zanella, R., Palli, G., Pirozzi, S., Melchiorri, C. (2019). Integration of robotic vision and tactile sensing for wire-terminal insertion tasks. IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, 16(2), 585-598 [10.1109/TASE.2018.2847222].

Availability:
This version is available at: https://hdl.handle.net/11585/685703 since: 2019-04-30

Published:

DOI: http://doi.org/10.1109/TASE.2018.2847222

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

16 September 2026

This is the final peer-reviewed accepted manuscript of:

D. De Gregorio, R. Zanella, G. Palli, S. Pirozzi and C. Melchiorri, ***"Integration of Robotic Vision and Tactile Sensing for Wire-Terminal Insertion Tasks****" in IEEE Transactions on Automation Science and Engineering, vol. 16, no. 2, pp. 585-598, April 2019*

The final published version is available online at:
https://doi.org/10.1109/TASE.2018.2847222

Rights / License:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/)

When citing, please refer to the published version.

Integration of Robotic Vision and Tactile Sensing for Wire-Terminal Insertion Tasks

D. De Gregorio[1], R. Zanella[1], G. Palli[1], S. Pirozzi[2] and C. Melchiorri[1]

[1]DEI - Università degli Studi di Bologna, 40136 Bologna, Italy
[2]DIII - Università degli Studi della Campania, 81100 Caserta, Italy

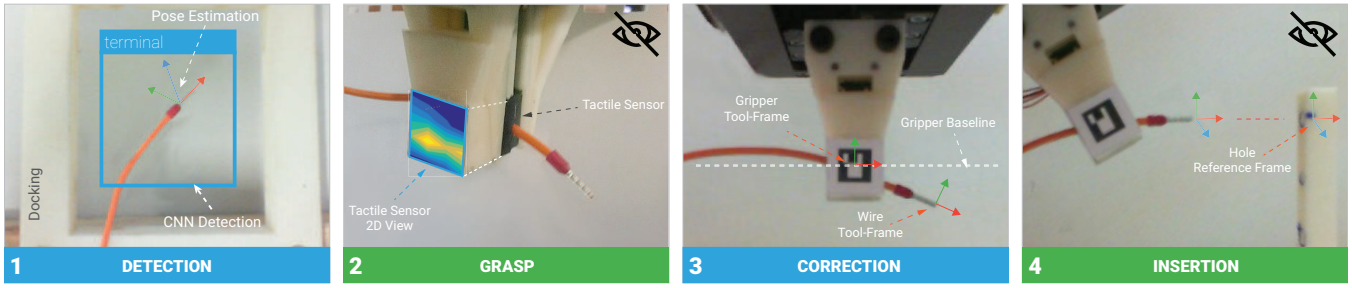


Fig. 1. The system's pipeline is composed by four components: 1) The wire detection, that exploits a self-trained multiple CNN triangulation for pose estimation; 2) The wire grasp during which a tactile sensor is used together with MLP to classify the grasp; 3) The pose correction of the wire; 4) The wire insertion into the terminal hole exploits the tactile sensor and a MLP trained to detect collisions.

***Abstract*— This paper reports the development of a manipulation system for electric wires, implemented by means of a commercial gripper installed on an industrial manipulator and equipped with cameras and suitably designed tactile sensors. The purpose of this system is the execution of wire insertion on commercial electromechanical components. The synergy between computer vision and tactile sensing is necessary because, in a real environment, the tight spaces very often prevent the possibility to use the vision system, also when the same task is performed by a human being. A novel technique to speed up the generation of training datasets for Convolutional Neural Networks (CNN) is proposed. Therefore, this technique is used to train a CNN in order to detect small objects (like wire terminals). Moreover, aiming to prevent faults during the task and to interact with the environment safely, several machine learning approaches are used to produce an affordable output from the tactile sensor. The proposed approach shows how a cheap sensor embedded with suitable intelligence can provide information comparable to a more expensive force sensor.**

***Index Terms*— Machine Learning, Robotic Vision, Tactile Sensors, Dexterous Manipulation, Deformable Objects.**

***Note to Practitioners*— This paper was motivated by the lack of commercial solution for the automatic cabling of switchgears. Existing approaches to this problem are in some way limited to specific large scale products or simple layouts. This paper investigated a robust and flexible solution, based on the exploitation of multiple sensors and machine learning algorithms, for wire detection, grasping and connection. The proposed approach is characterized by simple design and self-tuning capabilities, and it can be easily employed on a wide range of switchgear layouts thanks to the large workspace of the manipulator. Experimental results show that the proposed system is able to achieve a 95% success rate within a realistic admissible region. In future research, we will integrate the proposed solution with an electromechanical component localization module and a terminal fastening system to evaluate the performance on the real production line.**

I. Introduction

Switchgears and control panels are basic components of power generation, transformer and distribution stations, commercial and institutional buildings, industrial plants and automated factories, automatic machines and civil houses. In the actual scenario, the switchgear wiring is mostly executed by human operators because of the complex manipulation tasks, the large variability of the design, usually characterized by highly-customized solutions and small lot or, more frequently, single item production.

On the other hand, for series production of small-sized switchgears, a couple of commercial solutions for automatic wiring are available on the market. The SYNDY robotized wiring tool [1] requires a relatively large space between the components in the switchgear, mainly because of the dimensions and the mobility requirements of the end effector. It results that SYNDY is not suited to operate with a large number of wires or with high component density, fact that limits its applicability to the wiring of ceiling lamps. Another solution is the Kiesling Averex wiring [2], but it can operate only with fixed orientation of the wire and the screw, it presents a limited workspace and requires the switchgear to be in horizontal position.

Generally speaking, the wiring is composed by a sequence of single wire connections. Each wire connection implies:

A) the localization in the switchgear of the components to be connected, i.e. the points in which the two wire ends will be placed (since their position is known with some uncertainty);
B) first wire end connection;
C) wire routing inside the wire collector;
D) second wire end connection.

Step A), that is out of the scope of this paper, can be executed once at the beginning of the whole process, while the other steps from B) to D) must be repeated for all the connections in the switchgear net list. Moreover, step B) can be further decomposed in the following phases: 1) wire

localization; 2) wire grasping; 3) wire pose detection and correction; 4) insertion into the terminal; 5) tightening of the terminal screw (that can be achieved by a screwdriver or is automatic depending on the terminal type); 6) wire connection check. In particular, this paper deals with the phases from 1) to 4), see Fig. 1 where the implemented task sequence is depicted, while the following phases 5) and 6), as well as step C) and D), will be object of future research. In this paper, we suppose that the wires are prepared in advance by a devoted automatic machine. However, even if this paper focuses on a portion of the complete switchgear cabling, the implementation is carried out taking the whole process in mind. Even a single wire connection is really a challenging task for a robotic system, since it can be seen as a sub-millimeter precision peg-in-hole problem involving the manipulation of a deformable object, i.e. the wire. Moreover, the terminal position, or connection point, is usually known only approximately since the components are mounted on DIN rails, therefore a certain mobility along the rail is possible. The terminals are also difficult to see and to access, due to their location on both sides of the electromechanical components, the proximity of other components and wire collectors, and the presence of previously connected wires. These issues limit the applicability of vision systems to guide and control the wiring process. Therefore, in the system implementation we considered the requirements and the constraints imposed by the whole process, the space limitations imposed by the specific application, as well as the overall system cost and complexity.

In [3], preliminary results obtained within the WIRES experiment are presented. The project aims to automatize the switchgear wiring process. This objective is pursued by adopting industrial manipulators and properly design hardware and software tools. In [4] the design of the tactile sensor developed for this experiment is presented. This sensor is able to estimate both position and orientation of the grasped wire with respect to a reference frame placed at the center of the tactile sensor. In this work, the problem of wire grasping and the insertion of one wire end into a component terminal is addressed, supposing that the position of the terminal is known. To this end, we will present, for the first time to the best of our knowledge, how the vision and tactile feedback are complementary exploited and combined with several machine learning approaches to solve the problem of wire detection, manipulation and insertion into the terminal hole. It is worth mentioning that, even if no occlusion given by other wires or components is present in the experimental setup, the gripper adopted to grasp the wire prevents by itself the usage of a vision system to provide a close view of the terminal from front. It will be shown that the features and characteristics of the vision and tactile sensors are complementary for the task at hand. As a matter of fact, they both provide information that are strictly needed to achieve the required precision. This is due by the fact that, in a realistic scenario, the tight spaces between wires and components prevent the use of the vision system, in particular during the final part of the wire insertion, also when the same task is performed by a human being. It results that, in most of the practical cases, when a sensor can generate useful information, the other can not and vice versa. In the insertion phase, the tactile sensor can be used to quantify a collision, in order to evaluate if the insertion is correctly achieved. Moreover, in this work a novel technique to automatize and speed up the generation of training datasets is presented. This technique is exploited to train a Convolutional Neural Network (CNN) in order to detect small objects (like wire terminals), and a new method to estimate their 3D positions using multiple CNN predictions is shown. Additionally, Multi-Layer Perceptrons (MLPs), Random Forests (RFs) and Support Vector Machines (SVMs) are either trained to produce an affordable output from the tactile sensor to evaluate the correctness of the wire insertion task and detect faults. The application considered in this paper implies several issues related to micro-assembly processes, manipulation of deformable objects, occlusion in computer vision systems, tactile sensing, vision-tactile fusion and machine learning. Therefore, this application is of great interest for the overall robotic community.

The paper is organized as follows. Section II reports a summary of the previous researches carried out by other authors in this field. Section III introduces an high-level vision of the wire insertion problem. Section IV presents the hardware setup exploited during experiments. Section V describes in detail the components of the system pipeline. Finally, Sec. VI reports the set of experimental tasks that show the effectiveness of the proposed approach.

II. Related Works

A number of previous research activities can be found in literature about the modeling, the manipulation and the visual tracking of Deformable Linear Objects (DLOs) such as electric wires, demonstrating the large interest in this field.

In [5] a method to calculate the force acting on a purely-elastic flexible wire from its shape observed by stereo vision is developed. The same authors presented a method to insert a purely-elastic flexible wire into a hole observing the shape of the wire by stereo vision in [6]. The task of picking up cables from approximately known positions with an industrial robot using two light barriers has also been investigated in [7]. In [8], [9] the authors presented the static and the dynamic modeling of DLOs based on differential geometry coordinates, respectively. In [10] a path planning algorithm for DLOs subject to manipulation constraints is presented. In [11] a motion planner for manipulating DLOs and tying knots using two cooperating robotic arms is developed. In [12] a DLOs model based on mechanically rigorous and geometrically exact dynamic splines including both elastic and plastic deformation is described. In [13] a modeling of electric cables based on the visual measurement of their static and dynamic deformation is performed for cable insertion in electric and automotive industries. In [14] an algorithm for tracking DLOs based on a probabilistic generative model that incorporates observation and the physical properties of the tracked object is presented. In [15] the manipulation

planning problem of a DLO handled by a gripper at one of its extremities in free or contact space is considered. In [16], the robot is guided to grasp the wire on the clamp cover adopting a SIFT (Scale-Invariant Feature Transform) based algorithm. The problem of assembling flexible wire harness into instrument panel frame is addressed in [17] by making use of vision sensors and markers attached on the surfaces of the clamp.

Tactile sensing and vision are two synergistic modalities for manipulation. Vision systems provide rich information regarding unknown objects, in fact they became one of the main feedback source in robotics. However, they are often difficult to be applicable when the objects are occluded or visually confused. Recent progress in artificial touch sensing hardware allowed the robotics community to endow robots with touch capabilities and to show that tactile sensing can be efficiently employed in robot grasping. In order to deal with complex tactile information, machine learning algorithms have been widely used to address the classification problem. A grasp detection deep network is proposed in [18] to detect the grasp rectangle from the visual image with a new metric to assess the stability of the grasp. In [19] a novel method to systematically solve the visual-tactile fusion in object recognition tasks using multivariate time series is developed. In [20] visual modality is used to aid learning tactile modality during the training phase. In [21] the authors propose a cross-modal approach based on the use of visuo-tactile data for object recognition. A comparative analysis of classification algorithms for tactile sensors mounted on humanoid hand is presented in [22]. In [23] they present a robotic agent that learns to derive object grasp stability from touch. Classification is conducted through kernel logistic regression, applied to a low- dimensional approximation of the tactile data read from the robots hand. The implementation of tactile object identification and feature extraction techniques is discussed in [24], where two methods of tactile data interpretation are combined on data acquired during a single unplanned grasp: a random forests classifier and parametric object property estimators.

In this paper, our aim is to combine vision, tactile sensing and machine learning to manipulate electric wires and insert them into electromechanical components, taking into account the real manufacturing application constraints. This work will be part of an automatic switchgear wiring system under development. Previous approaches to similar manufacturing problems are mainly based on vision. The strength of the approach presented in this paper relies mainly on the synergistic combination of vision and tactile data to overcome the application constraints. The way how to combine these sensors has been selected taking into account that there are working conditions in which one of these two sensors can be ineffective or unreliable. Moreover, the whole development is performed trying to reduce complexity and cost of the final system.

This manufacturing application is really relevant in the industrial scenario, since the switchgear wiring still today represents a completely manual operation, that in turn results the major cost in the production of these highly customized items. Moreover, the interest in this field is confirmed by the number of literature works and companies involved. The proposed work is relevant because first there is no literature dealing with the overall task sequence, secondly it investigates a novel solution based on the combination of different technologies, i.e. vision, tactile sensing and machine learning to solve a problem that remains unsolved in the actual industrial practice.

The assumptions considered in this work are the following:

- only a limited part of the complete wiring task is taken into account to simplify the analysis;
- since the system is designed for an industrial setting, we suppose to operate in a partially structured environment;
- the resolution of the vision system and tactile sensor is somehow limited by the use of low-cost devices.

Besides this latter point is a benefit form the point of view of the overall system cost, resolution problems are mitigated by the introduction on suitable techniques, such as the exploitation of multiple views and machine learning, to achieve the desired task success rate.

III. Task Description

The task considered in this work consists in the insertion of an electric wire terminal in a hole that emulates the electromechanical component connector. Since it is really difficult to evaluate the correct alignment and the contact of the wire terminal using a real electromechanical component, an emulation body composed by a beam with a 5 mm pass-through hole has been used to ease the evaluation of the system performance without affecting results. With reference to Fig. 1, the task to be executed by the robot is composed by the following operations: 1) Detect the pose of the wire terminal using vision feedback in order to grasp it; 2) grasp the wire and validate the grasp through tactile feedback to evaluate if the following steps can be correctly executed; 3) Estimate the pose of the wire end w.r.t. the gripper with the required precision using vision feedback in order to correctly execute the insertion task; 4) Execute the wire insertion into the terminal hole detecting possible collisions by means of tactile feedback.

In the 2nd frame of Fig. 1 the *blind icon* is shown to emphasize that the vision system can't be used for monitoring the scene in this phase since the grasp, obviously, needs to be monitored by a device able to detect the physical contact with the object, e.g. the tactile sensor. The same holds for the 4th frame, showing the wire insertion, since in the real scenario the limited space available and the presence of other components and wires in the neighborhood of the working region prevent the scene observation by the camera, because both of occlusion problems and the impossibility of placing the camera close to the fingers. It results that the designed tactile sensor only fits with the available space in the region close to the wire connection point in the application scenario. This assumption is clearly not true in our experimental setup created ad-hoc to evaluate the effectiveness of the system,

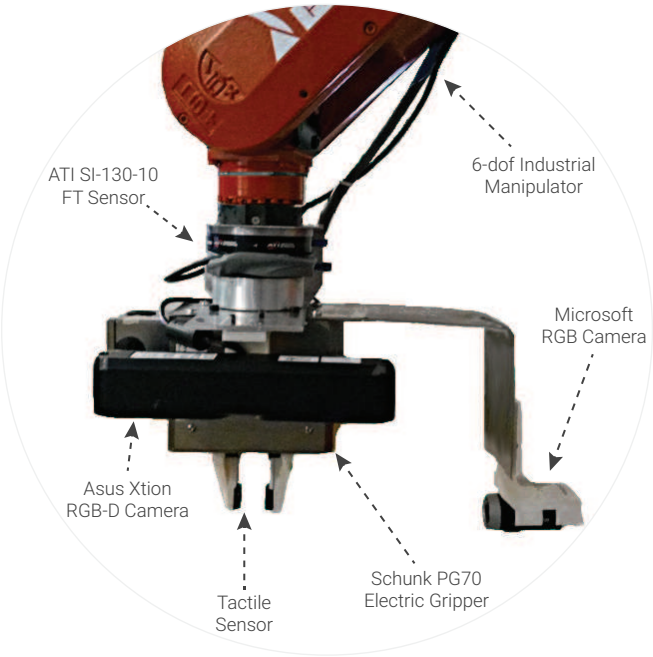


Fig. 2. The end effector used during the experiments. The Microsoft RGB Camera (referred as *side Camera*) is mounted on the end effector for evaluation purposes. In the real scenario the *side Camera* will be fixed to ground, to reduce the end-effector encumbrance.

but we selected to not use the vision during the insertion to recreate the real working conditions.

IV. The Experimental Setup

The hardware setup used during the experiments here described is shown in Fig. 2. The system is composed by an industrial manipulator, a COMAU Smart Six, equipped with a commercial gripper, a Schunk PG70 electric parallel gripper and an ATI SI-130-10 Force Torque (FT) sensor mounted on the wrist (between the robot interface and the gripper). Moreover, an Asus Xtion 3D camera with VGA resolution is mounted on the one gripper side pointing downward, to provide a top view of the scene (namely, the *hand camera* in the following). The 3D feature of this camera is useful for the reconstruction of the component location and encumbrance in the switchgear (these problems are not treated in this paper), but the 3D resolution is too poor for the purpose of wire terminal detection and grasping. Therefore, a computer vision algorithm has been developed, as reported in Sec. V-B, for reconstructing the 3D pose of the wire using multiple RGB images only provided by this camera.

On the other hand, to provide a close view of the task execution, an additional Microsoft 2D LifeCam camera with HD resolution is mounted on one gripper side (namely, the *side camera* in the following). In the experiments here reported, this camera was used also to estimate the wire pose after the grasp, as detailed in Sec. V-D. In normal conditions, this operation is performed by a fixed camera placed in a known position reachable by the robot to reduce the end-effector encumbrance.

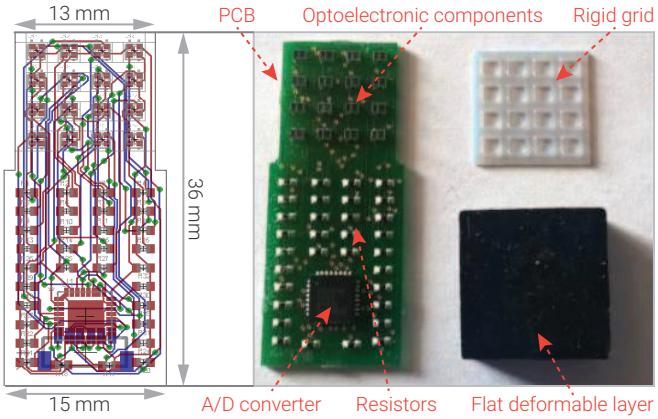


Fig. 3. Layout of the tactile sensor PCB on the left and pictures of the sensor components on the right.

A custom tactile sensor [3], [4] has been developed on the base of the one presented in [25] for the task here considered, see Fig. 3, and it has been mounted on one gripper's finger to provide a tactile image of the grasped objects, e.g. the wire. It is constituted by 16 taxels organized as a 4×4 matrix and a deformable layer with a flat shape. Each taxel is constituted by a single SMT photo-reflector integrating both an infrared LED and a PhotoTransistor (PT). When a contact with the deformable layer occurs, it produces vertical displacements of the reflective surfaces of the cells for all taxels. These displacements produce variations of the reflected light and, accordingly, of the photocurrents measured by the PTs. The taxel signals are acquired by a 16 channels ADC with Serial Peripheral Interface (SPI). The mechanical properties of the silicone cap determine the maximum load applicable to the sensor before cell saturation and, as a consequence, its sensitivity. The implemented sensor uses a shore hardness of 26 A, resulting in a maximum applicable force up to 30 N, with a sensitivity of about 0.3 N. An Arduino-based μcontroller board is then used to send the data to the control PC via USB connection.

The control system has been developed exploiting the ROS middleware to allow the communication between the different parts (sensors, robot, gripper, cameras, etc.) that compose the experimental setup.

Despite the gripper previously described is a preliminary solution, the end effector that will be adopted for the implementation of the whole cabling process is much more complex, see Fig. 4. The whole end effector will integrate an FT sensor at the wrist interface, a 3D camera providing top view of the scene, an computer-controlled screwdriver (for the execution of phase 5)) and a 4-DOFs gripper (gripper opening and finger x-y-z position w.r.t. to the screwdriver tip) equipped with the aforementioned tactile sensor. In the final process implementation, the robot arm will be used to position the screwdriver tip on the terminal screw, and the FT sensor will be used to control the contact with the screw during the tightening. Therefore, the end effector will be held in an almost fixed position, just the screw motion during the tightening will be compensated. Consequently, the wire

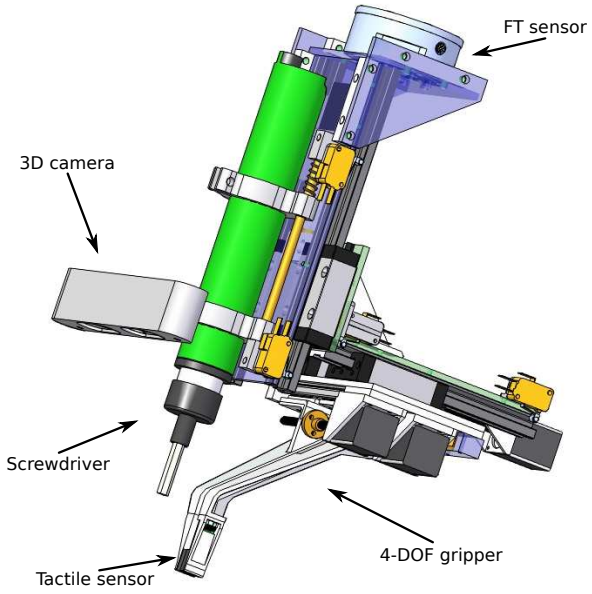


Fig. 4. CAD of the final end effector developed for the WIRES project.

Fig. 5. Different views of a switchgear during the assembly. The cables and components, often very similar and of the same color, make complex the detection of a specific terminal or wire from the scene.

insertion (phase 4)) will be performed by using the gripper DOFs only. It results that the FT sensor cannot be used during the insertion and for the tightening check, because the magnitude of the force generated by the wire contact during phase 4) is much lower than the one generated by the contact between the screwdriver and the screw, making the former indistinguishable. In this paper, the FT sensor has been used as ground through to train the tactile sensor integrated into the fingertip, as described in Sec. V-E.

Moreover, while the vision system can be easily applied during phase 1) and 3), it would be complex to adopt vision during phase 4) and 6) due to occlusion problems. In fact, several wires and components are usually present in the same scene, as shown in Fig. 5. Moreover, even if a camera will be mounted in the end effector, the field of view of this camera will provide a top view of the cables and components, for the execution of step A) and phase 1), while during phase 3) the point of view of this camera will be ineffective. Additionally, during phase 4) the wire insertion point (i.e. the terminal) will likely be occluded by the component itself (since the terminals are usually located on two opposite sides of the component). The use of a stereo or 3D camera for phase 3) was discarded because, due to the relative position of the wire tip and the camera, the vision system must be explicitly designed for this phase to achieve the required precision and range of view. It follows that the same camera cannot be used for other operations such as step A) that requires a much larger vision field. Therefore, we opted for a 3D camera providing a top view of the scene with a vision field selected mainly for step A). In the final setup, a second 2D fixed camera will be placed close to the wire picking point (and not mounted on the end effector as in Fig. 2). After phase 2), the robot will place the gripper in front this fixed camera to obtain a lateral view of the wire to execute phase 3). This solution do not increase the end-effector complexity, can be easily implemented since no space restrictions are present in the neighborhood of the wire picking region, is cheap and provide the better point of view to correct the most likely wire misalignment, i.e. in the gripper grasping plane.

V. System Pipeline

In this section, the self-labeling of the vision system training dataset by the robot will be introduced first, that represents the main novelty of the developed system. Secondly, the whole task execution pipeline as depicted in Fig. 1 will be presented. Each of the four modules will be described in detail, highlighting the adopted machine learning algorithms along with the related training techniques.

An overview of the pipeline is depicted in Fig. 6. In this picture the system is presented with the flowchart metaphor to understand better the dataflow and the interconnections between subsystems. It should be noted that the *self-training* procedure, described in Sec. V-A, is not present because technically, as described later, it is functionally equal to the procedure described in Sec. V-B and, moreover, it is an offline procedure not strictly related to the online pipeline.

A. CNN Self-Training for Wire Terminal Detection

The major novelty of the vision system here developed is the self-learning procedure exploited by the robot to train the CNN for wire terminals recognition. A popular approach introduced in Mitash *et. al.* [27], that uses synthetic data (physics emulation), is not suitable for deformable objects like wires. Also the approach proposed by Georgakis *et. al.*

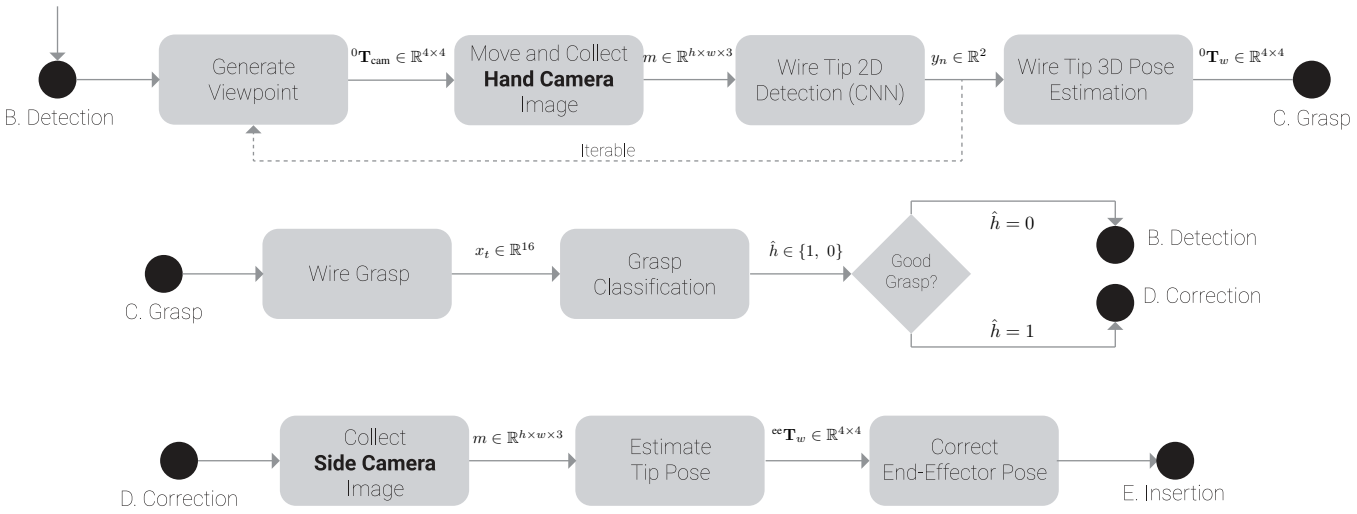


Fig. 6. The overall Pipeline of our System resumed as an High Level Flowchart. The system is split into 3 sub-systems to facilitate its representation and labeled edges are depicted to understand better the dataflow. Each subsystem is described in detail in the related subsection: *B-Detection* in the Sec. V-B; *C-Grasp* in the Sec. V-C; *D-Correction* in the Sec. V-D and *E-Insertion* in the Sec. V-E (the latter is not represented as a detailed flowchart module because its simple nature, and furthermore it's the topic of a future research. The notation $m \in \mathbb{R}^{h\times w\times3}$, introduced in this graph, represents a generic 3-channel image (e.g. an RGB image).

[28] cannot scale enough to industrial environment due to the lack of public datasets outside the service robotics field. In our approach, a CNN [26] was trained to detect square-regions around the wire end in RGB images gathered by the hand camera (the camera mounted on the end effector). The adopted CNN also discerns good terminals from bad ones (e.g. bad-crimped ferrules). This CNN differs from a classical Region Based CNN (R-CNN) [29] because reframes the problem of detecting a square region and, at the same time, classifies the object within as a unique regression problem.

The CNN is not trained from scratch, instead a *Fine-Tuning* procedure is performed starting from the original network [26] pre-trained over the *ImageNet*[30] dataset. We trained the network for 100000 steps with a *batch size* of 24. In Fig. 7(a) an example of self-labeled entries is presented. Our experimental dataset contains over 5000 RGB images, built in 4 hours (0.5 hours of human work + 3.5 hours of robot work). Considering an average time of 1 hour per 100 images, (estimated during our observations), the same labeling procedure done by hand would last at least 50 hours. With the term *labeled* image, we mean an image with square regions drawn around target objects used to train a CNN to detect the same object/region in new unseen pictures. As for other machine learning approaches, also in this case the larger is the number of training samples, the better is the CNN recognition rate.

The solution here proposed exploits the inverse technique shown in Fig. 8, that will be detailed in the next subsection. Given the position $\mathbf{p}_n$ of the wire terminals in homogeneous coordinates w.r.t. the robot reference frame, e.g. through measurement or by touching them with the end effector (this is the only human intervention), for every picture taken with the hand camera, it is possible to compute the projection y_n of $\mathbf{p}_n$ in the image coordinates using the pinhole camera equation

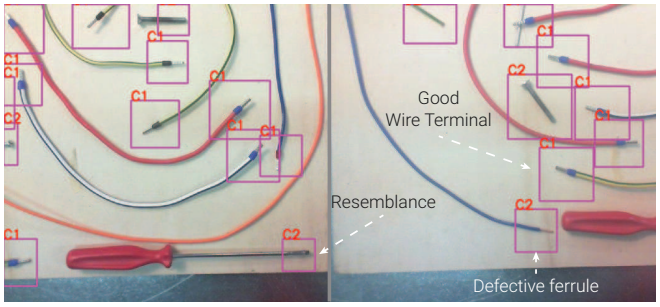
$$y_n = A \begin{bmatrix} R_{\text{cam}} & t_{\text{cam}} \end{bmatrix} \mathbf{p}_n, \quad A = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

where A represents the intrinsic camera matrix (a camera dependent parameter). The matrix $\begin{bmatrix} R_{\text{cam}} & t_{\text{cam}} \end{bmatrix} \in \mathbb{R}^{3\times4}$ represents the poses of the hand camera in the world coordinate frame (i.e. the camera extrinsic parameters). This matrix can be computed given by the position of the robot end effector (provided by the robot) and the relative position between the end effector and the hand camera (known from the end-effector design). In this way, given a scene with cables in known positions, it is possible to collect an arbitrary number of self-labeled training images just moving the hand camera around using the robot.

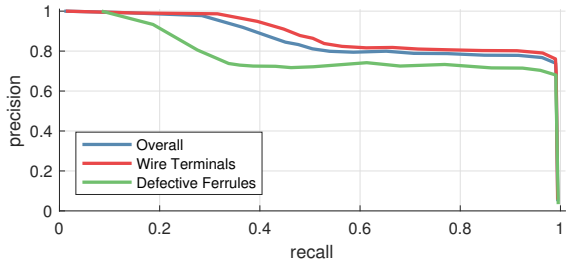
The outcome of this CNN is a set of rectangular frames, i.e. the "predictions", identified by both their 2D center coordinates x, y in the image plane, width w and height h, along with a label l identifying good ($l = 1$) or bad terminals ($l = 0$).

The Fig. 7(b) shows Precision-Recall curves for the object detector used during experiments (YOLO [26]). The mean Average Precision (mAP) for the overall detection task is 0.854, with a 0.88 mAP for wire terminals and 0.77 mAP for defective ferrules. This precision gap is justified on the grounds that a wire terminal is higher in visual cues w.r.t. a simple defective ferrule. Considering that:

- the human intervention takes only 30 minutes;
- this labeling procedure needs to be repeated whenever the environmental conditions change or a new class of Wire Terminals is provided;
- in an industrial setting, the environmental conditions can be controlled and maintained fairly constant;



(a) A couple of self-labeled random frames. This procedure generates more than *5000* labeled images in approximately *4 hours*. The same labeling task if had been entrusted to a distributed service like *Amazon Mechanical Turk*, minimizing user rewards, it would cost around *1000$*. Our labeling procedure trains the network to distinguish between the wire terminals (*Class1*=*C1*) and defective ferrules, or other resembling things (*Class2*=*C2*).



(b) Precision-Recall curves of the YOLO Object Detector ([26]) varying the output threshold. The mean average precision is 0.854.

Fig. 7. Qualitative and quantitative results for the self-labeling procedure. The presented approach provides high performance with minimal human intervention.

- just two classes of wire terminals are used in the actual industrial production;

the overall performance represents a suitable trade-off between functionality of the automatic system and required operator time. However, the proposed approach allows to easily generate training datasets in an extremely wide set of working conditions, far beyond the industrial scenario.

B. Wire Detection through CNN

It is worth mentioning that, in the real scenario, the wires to be connected into the switchgear will be produced and stored in a known region by an dedicated machine. Therefore, we can assume that the wires are arranged on a plane in such a way the gripper can grasp each wire without colliding with other wires or the environment. However, due to possible wire bends, the robot needs to locate the wire terminal and estimate its pose with sufficient precision to plan a correct grasp by using the hand camera.

This module exploits the output of the CNN described in the previous section to estimated the wire terminal 3D location in each image provided by the hand camera. This procedure is needed since the 3D hand camera resolution is not sufficient to achieve the desired wire grasp success rate. Thus, we need to reconstruct the 3D position of the target object, in this case the wire terminal, only exploiting multiple 2D information provided by the CNN. This is the

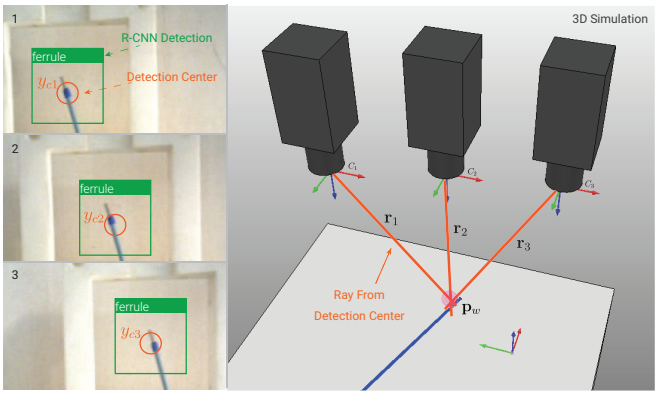


Fig. 8. Depth estimation through multiple CNN detections. Three images were captured from the camera poses $C_{\{1,2,3\}}$ respectively, obtaining 3 different square regions through the "predictions". By projecting the rays $\mathbf{r}_{\{1,2,3\}}$ passing through each region's center $y_{c\{1,2,3\}}$, the wire terminal $\mathbf{p}_w$ is found. Conversely, knowing the wire terminal position $\mathbf{p}_w$ in the robot coordinate system and moving the hand camera using the robot, images from an arbitrary number of known camera poses C_n can be collected. Therefore, $\mathbf{p}_w$ can be projected into the images coordinates $\hat{y}_{cn}$, generating an arbitrary number of self-labeled images for the CNN training.

well-known technique called Structure From Motion (SFM) [31], which exploits the triangulation algorithm to reconstruct the 3D coordinates of a 2D feature seen from multiple known vantage points. In the classical SFM approach, as well as in SLAM systems, the camera pose is computed simultaneously to the reconstruction phase. Unlike these approaches, we rely on the technique described in [32]. This technique exploits the robot high repeatability, along with a precise hand camera calibration, to compute in a closed form the 6-DOF pose of the hand camera. Theoretically, if we know the exact homogeneous coordinate y_n of our target object in the image I_n, two viewpoints only are needed to obtain a suitable triangulation result. However, in our case only a coarse region around our target terminal is available, as depicted in Fig. 8. If the center of the square-region is chosen as 2D reference feature, a non-perfect overlapping with the tracked object center is obtained (this error strictly depends upon CNN architecture and is not treated in this work). Therefore, more vantage points are needed to achieve a more accurate 3D reconstruction. For each vantage point, a ray $\mathbf{r}_n$ can be computed, i.e. a unit vector in the camera reference frame, corresponding to the selected 2D feature:

$$\mathbf{r}_n = \left\| A^{-1} y_n \right\| \tag{2}$$

Together with the center of the camera frame $\mathbf{c}_n$, $\mathbf{r}_n$ generates a 3D line $l_n = (\mathbf{r}_n, \mathbf{c}_n)$. In the camera reference frame, $\mathbf{c}_n$ is zero, otherwise it represents the position of the hand camera in the world coordinate frame. Thus, given a set of lines l_n, the closest point $\mathbf{p}$, i.e. the point with minimum distance from all the lines (since a common intersection point could not exist with real measurements) can be computed by

$$\mathbf{p} = \left(\sum_i \mathbf{I} - \hat{\mathbf{r}}_i \hat{\mathbf{r}}_i^T\right)^{-1} \left(\sum_i \left(\mathbf{I} - \hat{\mathbf{r}}_i \hat{\mathbf{r}}_i^T\right) \mathbf{c}_i\right)$$

where $\hat{\mathbf{r}}_i$ is any perpendicular unit vector to $\mathbf{r}_i$ [33].

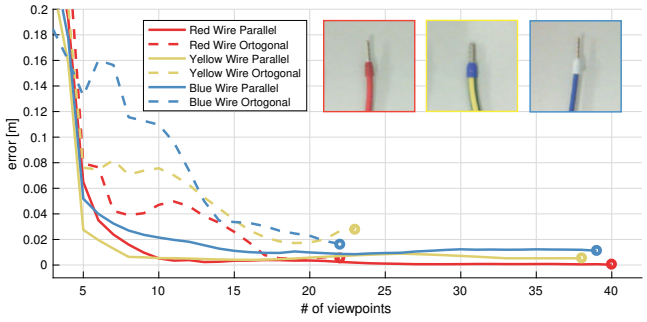


Fig. 9. Wire terminal depth estimation error w.r.t. the number of images collected from different viewpoints moving the hand camera by using the robot. Three wire terminals, *Red*, *Yellow* and *Blue*, and two motion directions, *Parallel* or *Orthogonal* to the plane on which the cable lies, were tested. The *Parallel* movement provides better results since it minimizes the likelihood between the rays, achieving an error lower than 1 cm after 10÷15 images.

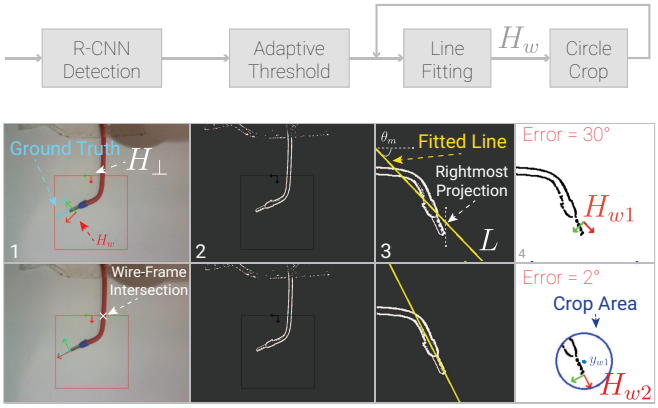


Fig. 10. 2D wire terminal reference frame estimation. Each row represents a pipeline iteration: *1)* CNN detection over the wire image; *2)* binary output obtained by the adaptive threshold; *3)* the detected region is rotated w.r.t. $H_\perp$ and the fitting line is superimposed; *4)* the wire reference frame H_{w1} is laid out. In the second row, another iteration is performed carving out a circular region around y_{w1} and repeating the line fitting process.

In Fig. 9 the results of the proposed algorithm scanning three different wires (in color and dimension) with different camera movements (parallel and orthogonal to the image plane) are presented. The achieved precision is approximately 1 cm collecting more than 20 images from different viewpoints. At the same time our results show that the best performance is achieved by moving the camera parallel to image plane. These experiments are designed as a proof of the approach correctness and not to evaluate best hyper-parameters to reduce the error (i.e. the optimal distance between camera and target object). To generalize the error in the depth estimation, the Stereo Vision error formulated by Gallip *et. al* [34] can be used

$$\epsilon_z = \frac{z^2}{b \cdot f}\epsilon_d$$

where ϵ_z is the depth estimation error, z is the distance, f is the focal length of the camera, b is the baseline (e.g. the distance between the two camera, in a classical stereo vision setup, or the two vantage points in our system) and e_d is the disparity error. Since the 6-DOF camera pose can be controlled by moving the manipulator, it is possible to select the couple (z, b) to reduce the error ϵ_z according to the circumstances. Now, we can define a generic function:

$$\tau(y_0, ..., y_n, C_0, ..., C_n) = \mathbf{p} \tag{3}$$

to compute the 3D position $\mathbf{p}$, corresponding to the 2D feature y using multiple images.

Unfortunately, just one 2D feature is not enough to infer a 3D reference frame associated with the wire terminal. Therefore, at least 2 features are needed to estimate a 3D vector corresponding to the final part of the wire by means of the eq. (3). In Fig. 10 the algorithm used to infer a 2D reference frame H_w of the wire terminal is shown starting from a square region of the image, in a nutshell: starting from the image 1) an adaptive threshold is applied to the target region obtaining in 2) a binary image enhancing wire's pixels and removing background; 3) the region is rotated w.r.t. a custom reference frame $H_\perp$ placed on the mid point of one side of the frame, chosen such that it is the nearest point to the intersection of the wire with the square region; 4) an *orthogonal regression* is applied to the binary image to estimate the best fitting line $L = \{(x, y) \mid y = mx + q\}$ onto wire points; use the rightmost wire's pixel projection onto line as the center of H_w and the angle of L as its orientation, i.e.

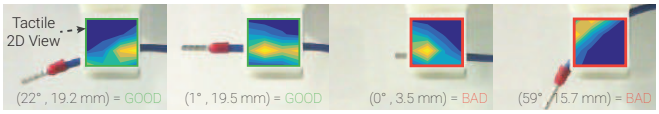
$$H_w = (y_w, \theta_w) = (y_w, \tan^{-1}(m)) \tag{4}$$

In Fig. 10 a case in which the cable is strongly bent is shown: this situation is useful to show the effectiveness of the proposed algorithm but it is unlikely in a real scenario. This sequence can be considered as the worst case, in which the ferrule is almost orthogonal to the portion of the cable intersecting the square region. The proposed algorithm produces a quite inaccurate wire end pose after the first iteration in this case. However, it is possible to see how a second iteration applied to a cropped region around the previous estimated center y_w ensures that line fitting is relative only to the terminal part and not to the whole wire within the region. In Sec. V-D the way how this algorithm is applied to the images provided by the side camera to estimate the wire end reference frame for pose correction will be further explained. Here, instead, we can just take advantage of the center homogeneous coordinate y_w of H_w and use eq. (3) to estimate its 3D position in the robot reference frame. Thus, from the pairs y_w, y_c, where y_c is the homogeneous coordinate of the center of the aforementioned CNN detection, see Fig. 8, collected from multiple viewpoints, we can compute their corresponding 3D points

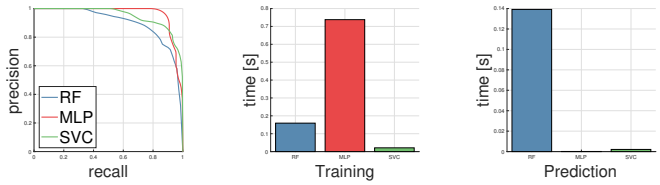
$$\mathbf{p}_w = \tau(y_{w\{1,...,n\}}, C_{\{1,...,n\}}), \;\; \mathbf{p}_c = \tau(y_{c\{1,...,n\}}, C_{\{1,...,n\}})$$

Thus, given the pair $(\mathbf{p}_w, \mathbf{p}_c)$,

$$\mathbf{v}_{w_x} = \frac{(\mathbf{p}_w - \mathbf{p}_c)}{\|\mathbf{p}_w - \mathbf{p}_c)\|}$$



(a) The binary classifier distinguishes between *Good* (1) and *Bad* (0) grasps using tactile sensor measurements.



(b) Classification metrics over RF, MLP and SVC. MLP shows better results and the lowest prediction time.

Fig. 11. The wire grasp classifier. This classifier was self-trained over the outcomes of the system module described in Sec. V-D choosing – by design – the thresholds that would affect the rest of the task (e.g. an escaping terminal shorter than 1 cm is considered a *bad* grasp).

represents the unit vector oriented along the wire terminal symmetry axis, while

$$\mathbf{v}_{w_z} = \mathbf{v}_{w_x} \times \pm\mathbf{u}_x, \;\; \mathbf{u}_x = [1 \quad 0 \quad 0]^T$$

indicates the forward direction w.r.t. the robot reference frame. Therefore, the pose ${}^0\mathbf{T}_w$ of the wire end can be defined as

$${}^0\mathbf{T}_w = \begin{bmatrix} \mathbf{v}_{w_x} & \mathbf{v}_{w_x} \times \mathbf{v}_{w_z} & \mathbf{v}_{w_z} & \mathbf{p}_c \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_w & \mathbf{p}_c \\ 0 & 1 \end{bmatrix} \quad (5)$$

Note that the orientation $\mathbf{R}_w$ is chosen by construction considering that the wire terminal is symmetric along its $\mathbf{v}_{w_x}$ axis. Thus, the $\mathbf{v}_{w_z}$ component, can be arbitrarily chosen to point toward the ceiling. To this end, the sign of $\mathbf{u}_x$ is chosen case-by-case to avoid that $\mathbf{v}_{w_z}$ points to the floor instead. The 6-DOF pose of the wire ${}^0\mathbf{T}_w$ is then used by the robot in order to perform the grasp.

C. Wire Grasp

In order to facilitate the insertion, the wire should be grasped at the center of the gripper fingers. It is important here to recall that, both in the real scenario and during the experiments here reported, the wire lays on a plane with its terminal section in the free space. This allows the gripper to grasp the wire without colliding with other wires or the environment.

Even though the module described in Sec. V-D can be used to detect if the wire as been effectively grasped or not, there will be no way to recover from a grasp failure at this stage, mainly because it will be very challenging to perform a re-grasp once the wire has been removed from its docking position. For this reason, the outcome of the wire pose detection module described in Sec. V-B is exploited to train a classifier able to detect if the cable is in a suitable pose w.r.t. the fingers from tactile sensor data only. Therefore, the use of the tactile sensor allows to evaluate immediately the grasp correctness, removing in this way the possibility of reaching unrecoverable situations.

Three different machine learning algorithms for classification are tested and compared: a Multi-layer Perceptron Neural Network (MLP) with 3 hidden layers composed of 16, 8 and 2 neurons; a Random Forest (RF) with 200 trees; and a Support Vector Classifier (SVC) with a radial basis function kernel. Figure 11 shows some example of *good* and *bad* grasps together with some classification benchmark among different classifiers trained during our experiments. The generic classifier is fed with $x_t \in \mathbb{R}^{16}$, representing the 4×4 matrix of the tactile measurement, coupled with $\hat{h} \in \{1, 0\}$, that is a boolean information representing if the wire grasp configuration is within an admissible range or not respectively. Taking into account eq. (4), the parameter $\hat{h}$ is defined as

$$\hat{h} = \begin{cases} 1 & l_{\text{min}} \geq \|\perp y_w\| \geq l_{\text{max}} \wedge \theta({}^{\perp}H_w) \leq \theta_{\text{max}} \\ 0 & \text{otherwise} \end{cases}$$

where $\theta({}^{\perp}H_w)$ means the orientation of the 2D reference frame ${}^{\perp}H_w$ (i.e. is H_w expressed in $H_{\perp}$) and $\theta_{\text{max}}, l_{\text{min}}, l_{\text{max}}$ are the parameters defining the terminal orientation/position admissible range with respect to the fingers. The results shown in Fig. 11(b) were obtained with a dataset of over 200 grasp samples. The best performance is provided by the MLP algorithm. However, since the purpose of this classifier is to ensure that *bad* grasps are detected, the objective is to maximize *precision* and not *recall*. Therefore, any of the evaluated classifiers can be used for this problem because all reach precision equal to 1 in the *precision-recall* curve. In case of bad grasp is detected, the gripper is retracted without removing the wire from its docking, and the procedure restarts from the wire pose detection. Increasing the number of viewpoints can be used in this case to possibly reduce the wire pose estimation error.

D. Wire Pose Correction

In this stage of the pipeline, the system aims to estimate the pose of the wire w.r.t. the gripper by means of the side camera framing laterally the fingers, as depicted in Fig. 12. This problem is a simplification of the one seen in the Sec. V-B. Indeed, here the pose of the side camera ${}^{\text{ee}}\mathbf{T}_{\text{cam}}$ w.r.t to the gripper is known by construction. Therefore, the pose of the side camera in world coordinates can be easily computed as ${}^0\mathbf{T}_{\text{cam}} = {}^0\mathbf{T}_{\text{ee}}\,{}^{\text{ee}}\mathbf{T}_{\text{cam}}$, where ${}^0\mathbf{T}_{\text{ee}}$ is the actual forward kinematics solution. In case a fixed camera is used to that purpose, the camera position ${}^0\mathbf{T}_{\text{cam}}$ is known and the camera position w.r.t. the end effector ${}^{\text{ee}}\mathbf{T}_{\text{cam}}$ can be compute inverting the previous formula. A mandatory step of this module is to calibrate correctly the pose of the side camera ${}^0\mathbf{T}_{\text{cam}}$ in the robot coordinate system. To perform a correct extrinsic calibration, the method seen in [32] is exploited by means of an *Augmented Reality* marker [35] printed in a known pose w.r.t. the fingers of the gripper. The marker attached to the back side of finger is visible in Fig. 12. This approach allows to calibrate the side camera pose on-line, enabling to use a moving camera instead of a fixed one (e.g. camera mounted on another robot).

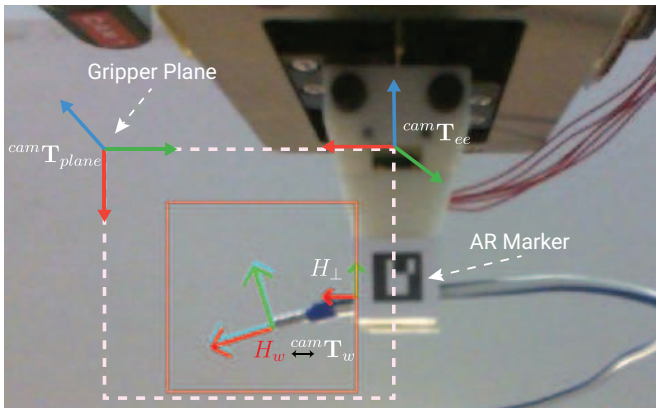


Fig. 12. Estimation of the wire terminal pose after grasp. The flattest part of the finger contains the *Augmented Reality Marker* used to calibrate the camera 6-DOF pose related to the robot reference frame.

By exploiting the knowledge of the camera and the gripper position, the distance of the grasp plane (i.e. the plane on which the wire lies after grasp) can be computed in analytical form as $d_w = {}^{\text{cam}}p_{z_{\text{ee}}}$, where ${}^{\text{cam}}p_{z_{\text{ee}}}$ is the z component of the translation part of ${}^{\text{cam}}\mathbf{T}_{\text{ee}}$. Then, given the *depth* d_w of the image pixels, the conversion from homogeneous to 3D coordinates (in the camera reference system) can be computed as

$$\pi(y) = \pi(\begin{bmatrix} u & v & 1 \end{bmatrix}^T) = \mathbf{p} = d_w \begin{bmatrix} \frac{(v-c_x)}{f_x} & \frac{(u-c_y)}{f_y} & 1 \end{bmatrix}^T \quad (6)$$

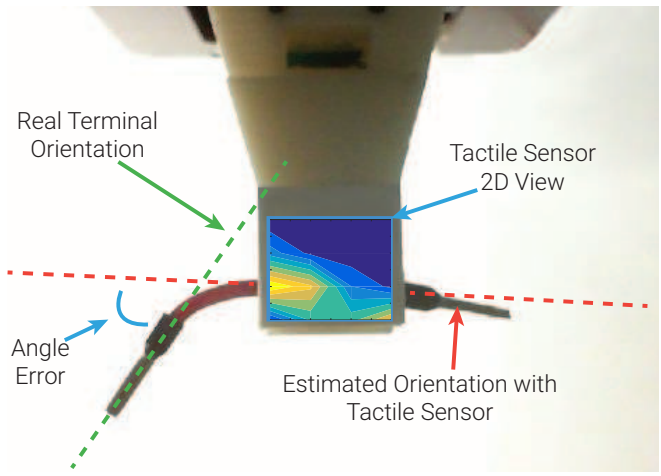
where $y = \begin{bmatrix} u & v & 1 \end{bmatrix}^T$ are the homogeneous coordinates of a generic pixel in the 2D image, $\mathbf{p}$ is the corresponding 3D point, c_x, c_y, f_x and f_y are the parameters of the camera matrix A. Thus, the systems exploits the same algorithm seen in Sec. V-B (see Fig. 10). The only difference is that, in this case, the reference frame $H_\perp$ is chosen to be as close as possible to the fingers' center. The whole procedure is shown in Fig. 12. Hence, given the position of the wire terminal w.r.t. the end effector ${}^{\text{ee}}\mathbf{T}_w$, the wire terminal position in the world coordinates is ${}^0\mathbf{T}_w = {}^0\mathbf{T}_{\text{ee}}\,{}^{\text{ee}}\mathbf{T}_w$.

In Fig. 13 the error rate on the 2D wire pose estimation after the grasp varying the tilt of the cable is reported. In Fig. 13(b) we can see how, choosing a desired *crop size e.g.* between 1 and 2, the estimation error is under 5 pixel for the position and 5 deg for the orientation (the dotted black line) considering a wire tilt angle in the range ± 45 deg (i.e. for the first three curves in the legends).

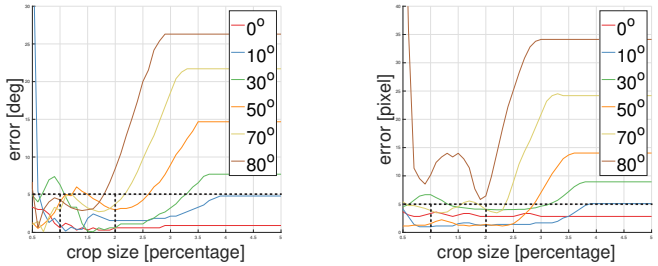
Obviously, the error metric derived from the pixel error is proportional to the distance between the camera and the target object; for a distance of 0.1 m we have a conversion factor of k = 0.0002, then 5 pixel = 0.0001 m error.

E. Wire Insertion

After the wire pose correction accomplished during the previous stage, the insertion task can be executed by planning a trajectory of the wire terminal frame ${}^0\mathbf{T}_w$ toward the component hole. This task can be seen as a *peg-in-hole* problem. In this phase, machine learning is exploited to



(a) 2D view of the tactile sensor is superimposed over a real RGB image of a grasped wire. The red dashed line represent the best fit provided by the tactile sensor.

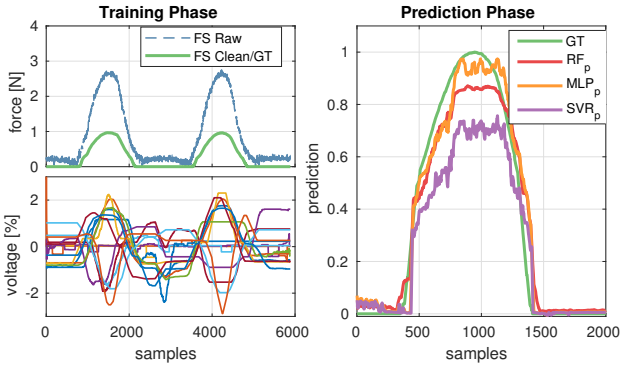


(b) Estimation error varying the angle of the wire terminal w.r.t. the prediction of the tactile sensor. The x-axis reports the crop radius normalized over the ferrule size (in pixel).

Fig. 13. Detection of the pose of a grasped wire by the vision system.

infer from the tactile sensor data the same information coming from the FT sensor. This is needed to detect impact between cable terminal and the component, and eventually to correct the wire trajectory during the insertion into the terminal. Normal and tangential forces components can be distinguished by the tactile sensor described in [25] since they are related, respectively, to symmetric and asymmetric variations of the measured pressure map. As a consequence, during a collision, a strong correlation exists between the direction of movement of the grasped object and the signal pattern x_t provided by the 4×4 taxels matrix. For this reason, x_t is exploited to train a regressor able to provide a scalar continuous variable representing the magnitude of the collision force in the tactile sensor plane.

Aiming at comparing alternative solutions, several regressors are trained by the robot itself, collecting data during many collisions between a flexible barrier and a grasped terminal in known pose (as provided by the wire pose correction module). These data are used to predict a real value associated with the impact force and quantify the latter in a continuous manner. In Fig. 14 the data used during the training procedure involving tactile and force sensor data are shown. From this figure, it is clear that the MLP produces a suitable prediction of the contact force. Figure 14 shows a Mean Square Error (MSE) and Dynamic Time Warping



	RF	**MLP**	SVR
MSE	0.0085	**0.0066**	0.0124
DTW	0.0264	**0.0182**	0.0347

Fig. 14. *Training* and the *Prediction* phases of the collision detection regressor. On the left plot, a couple of impacts measured by both the force sensor (top) and the tactile sensor (bottom); the green curve is the normalized output of the force sensor used as training reference. On the right plot, the execution of the predictors RF (Random Forest), MLP (Multi-Layer Perceptron) and SVR (Support Vector Regression) speculating about terminal collisions. On the bottom table, the performance of the predictors in terms of *MSE* (Mean Square Error) and *DTW* (Dynamic Time Warping [36]).

(DTW) analysis [36] performed over the *regressor*. The regressor is trained over 15000 samples collected during controlled impacts with different cable diameters in different angle w.r.t. the fingers. During training, the wrist-mounted FT sensor is used as ground truth. From these results, it is possible to conclude that the insertion task can be monitored exploiting the tactile sensor only, without using a vision system or a more expensive FT sensor.

A *Regressor* is adopted during the wire insertion to quantify the collisions instead of a simple *Classifier* that can simply detect it. By exploiting this approach, it is possible to distinguish between actual collisions, as shown in Fig. 16(a), from rubbing of the wire end inside the component terminal hole, as reported in Fig. 16(b). While in the former case the insertion task is stopped and completely replanned, in the latter one the task is very likely to be successfully accomplished. As a future development, we aim at exploiting the regressor output as a feedback signal to guide the insertion task. The aim is to increase the insertion success rate even in case of lateral contact between the wire end and the component terminal hole.

VI. Evaluation of the Insertion Task Sequence

In Fig. 15 a sequence showing the wire pose correction and insertion into the component simulacrum hole is shown. Starting from a common configuration (1), in the top sequence a wrong alignment is provided to the system at step (2) to shown the effect of collisions. It results that the wire terminal is not aligned with the reference line and, as a consequence, with the hole at stage (3), causing a collision at stage (4). In Fig. 16(a) the artificial wrong wire terminal orientation, the robot x-axis position and the filtered output of the MLP regressor over tactile sensor during the wire pose correction and insertion task are reported for the wrong sequence in Fig. 15. Looking in particular at the MLP output, the effect of the collision measured by the tactile sensor can be clearly seen in the final part of the insertion phase. The bottom sequence, instead, shows how the vision system allows to correctly align the wire terminal with the reference line at step (2) and, consequently, with the insertion hole at step (3). It results that the wire is correctly inserted into the hole at step (4). Figure 16(b) shows the wire terminal orientation, position and the MLP output over the tactile sensor data during the wire pose correction and insertion task reported in the good sequence of Fig. 15. After the initial estimation of the wire terminal pose, the end-effector orientation is corrected and the insertion task is executed: during the insertion, the MLP output allows to detect a contact between the wire terminal and the internal part of the hole causing friction. The output of the MLP regressor is not considered during the other phases to avoid false positive.

The performance of the overall pipeline were evaluated by executing the whole task for about 30 times with a wide range of working conditions. In Tab. I two subsets of 15 runs executed with two wires which external diameter is 2 mm and 3.5 mm, respectively, are reported. In these tables, m is the estimated wire terminal orientation, d denotes the distance of the wire tip from the finger center and c is 1 in case of successful insertion or 0 in case of failure. With reference to Fig. 16

$$c = \begin{cases} 0, & \text{Position} > 0.65 \wedge \text{MLP output} > 0.5 \\ 1, & \text{Position} > 0.65 \wedge \text{MLP output} < 0.5 \end{cases}$$

where the position threshold means that the wire terminal is inserted. The data reported in Tab. I include also experiments performed in extreme conditions to test the system robustness. It results a overall success rate of about 66%.

TABLE I

Wire insertion results for a cable with external diameter of 2.0 mm on the left and 3.5 mm on the right. Parameters m and d refer to initial condition of the wire w.r.t. the gripper, while c refers to the result of insertion where $c = 1$ is a positive outcome.

#	m [deg]	d [mm]	c	#	m [deg]	d [mm]	c
1	-5.7	48.0	1	1	13.5	27.0	1
2	-1.0	32.5	1	2	-20.3	30.0	1
3	4.0	38.9	1	3	4.6	29.4	1
4	-2.9	27.6	1	4	4.0	30.0	1
5	27.5	46.5	1	5	-12.4	29.0	1
6	-32.6	40.6	0	6	23.3	43.0	0
7	-15.6	38.5	1	7	-19.8	41.0	1
8	12.4	39.4	1	8	0.6	40.0	1
9	14.0	29.5	1	9	24.2	48.0	0
10	10.8	24.4	0	10	21.8	52.0	1
11	46.4	56.0	0	11	52.9	56.0	0
12	42.9	60.9	1	12	7.4	35.6	1
13	-28.4	45.3	0	13	47.5	52.0	1
14	41.3	65.9	0	14	58.6	95.0	0
15	33.0	39.0	1	15	44.7	69.2	0

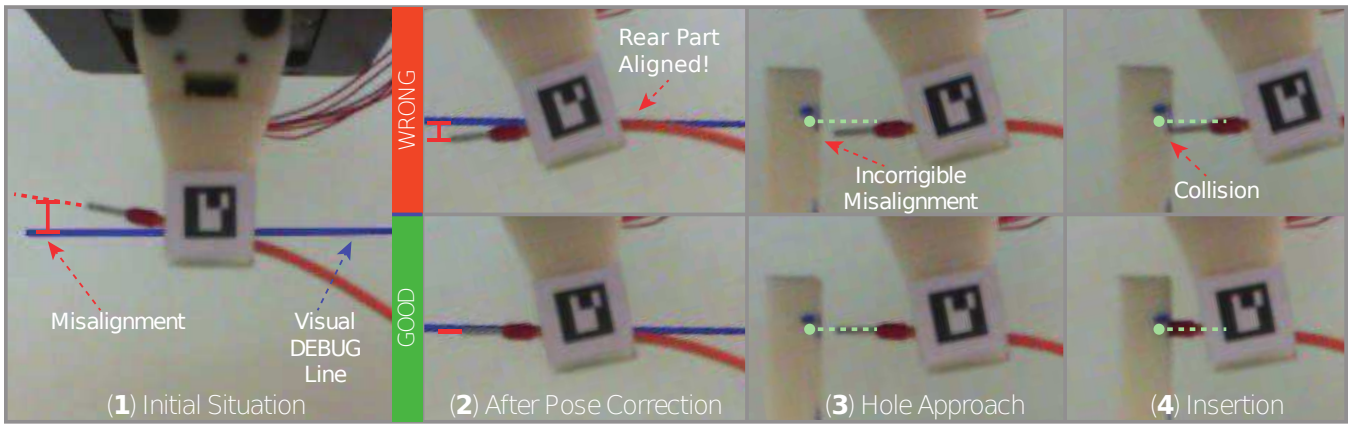
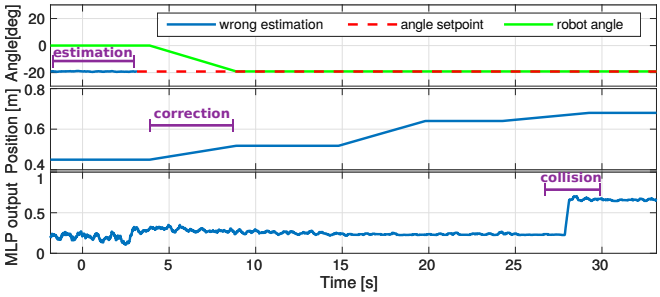
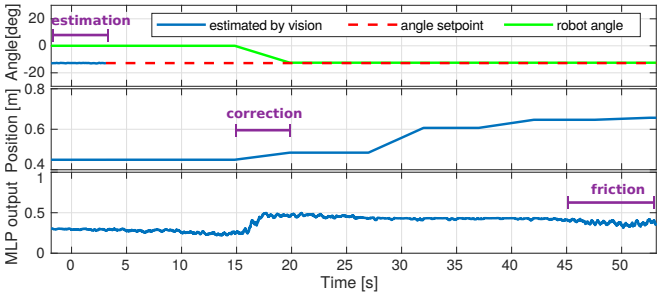


Fig. 15. Evaluation of the whole insertion task. Collision detection exploiting tactile feedback in case of wrong alignment (top sequence) and wire terminal pose correction and insertion using the vision feedback (bottom sequence).



(a) Collision detection using the tactile sensor.



(b) Wire pose correction and insertion using the vision system.

Fig. 16. Execution of the wire pose correction and insertion tasks.

Looking at these results in the $\{m, d\}$-plane, as reported in Fig. 17, it is possible to define an admissible working region of $m = \pm 20\,\text{deg}$ and $d \leq 50\,\text{mm}$ containing almost only successful wire insertions, 15 over 16, resulting in a success rate of about 95%. This working region can be easily addressed in the partially structured application scenario.

A qualitative evaluation for each building block of the insertion pipeline is shown in the supplementary material.

VII. Conclusions

In this paper, the system developed to perform wire detection, grasping and insertion into a component hole using suitable combinations of vision and tactile feedback is described. The synergy between cameras and tactile sensors allows to deal with the typical issues in the switchgear wiring scenario. Several machine learning algorithms are exploited

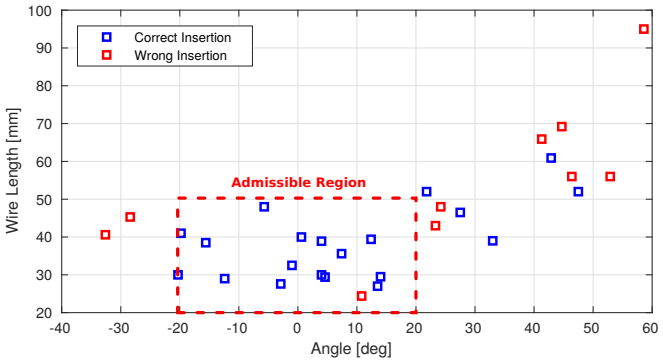


Fig. 17. Evaluation of the admissible operating range of the developed wire manipulation and insertion system.

for the system development, both for the vision and for the tactile module. Moreover, suitable techniques are developed for the automatic generation of the training datasets, allowing to significantly speed up the implementation of the target application. Future activities will be devoted to the integration with the component localization module and the terminal fastening system to test the overall wiring task execution pipeline. Moreover, the system evaluation in the real production scenario will be evaluated.

Acknowledgments

This work was supported by the European Commissions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 601116 (ECHORD++ project - WIRES Experiment).

References

[1] System Robot Automazione S.r.l., "The syndy robotized wiring tool," Online, http://www.systemrobot.it/syndy-en.asp.

[2] Kiesling Maschinentechnik, "Averex wiring centre," Online, https://www.rittalenclosures.com/averex/.

[3] M. Busi, A. Cirillo, D. De Gregorio, M. Indovini, G. De Maria, C. Melchiorri, C. Natale, G. Palli, and S. Pirozzi, "The WIRES experiment: tools and strategies for robotized switchgear cabling," in *Proc. of the 27th International Conference on Flexible Automation and Intelligent Manufacturing*, Modena, Italy, 2017.

[4] A. Cirillo, G. De Maria, C. Natale, and S. Pirozzi, "Design and evaluation of tactile sensors for the estimation of grasped wire shape," in *Proc. IEEE Int. Conf. on Advanced Intelligent Mechatronics*, Munich, Germany, July 3-7 2017, pp. 490–496.
[5] H. Nakagaki, K. Kitagaki, T. Ogasawara, and H. Tsukune, "Study of insertion task of a flexible wire into a hole by using visual tracking observed by stereo vision," in *Proc. of the Int. Conf. on Robotics and Automation*, vol. 4, 1996, pp. 3209–3214.
[6] ——, "Study of deformation and insertion tasks of a flexible wire," in *Proc. of the Int. Conf. on Robotics and Automation*, vol. 3, 1997, pp. 2397–2402.
[7] A. Remde, D. Henrich, and H. Wörn, "Picking-up deformable linear objects with industrial robots," 1999.
[8] H. Wakamatsu and S. Hirai, "Static modeling of linear object deformation based on differential geometry," *The International Journal of Robotics Research*, vol. 23, no. 3, pp. 293–311, 2004.
[9] H. Wakamatsu, K. Takahashi, and S. Hirai, "Dynamic modeling of linear object deformation based on differential geometry coordinates," in *Proc. of the Int. Conf. on Robotics and Automation*, 2005, pp. 1028–1033.
[10] M. Moll and L. E. Kavraki, "Path planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 625–636, 2006.
[11] M. Saha and P. Isto, "Manipulation planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1141–1150, 2007.
[12] A. Theetten, L. Grisoni, C. Andriot, and B. Barsky, "Geometrically exact dynamic splines," *Computer-Aided Design*, vol. 40, no. 1, pp. 35–48, 2008.
[13] Y. Kataoka and S. Hirai, "Vison-guided individual modeling of bendable cables for their insertion," in *2010 World Automation Congress*, 2010, pp. 1–8.
[14] J. Schulman, A. Lee, J. Ho, and P. Abbeel, "Tracking deformable objects with point clouds," in *Proc. of the Int. Conf. on Robotics and Automation*, 2013, pp. 1130–1137.
[15] O. Roussel and M. Taïx, "Deformable Linear Object manipulation planning with contacts," in *Robot Manipulation: What has been achieved and what remains to be done? Full day workshop at Int. Conf. on Intelligent Robots and Systems*, Chicago, United States, 2014.
[16] K.-m. Koo, X. Jiang, K. Kikuchi, A. Konno, and M. Uchiyama, "Development of a robot car wiring system," in *Proc. of the Int. Conf. on Advanced Intelligent Mechatronics*, 2008, pp. 862–867.
[17] X. Jiang, K.-m. Koo, K. Kikuchi, A. Konno, and M. Uchiyama, "Robotized assembly of a wire harness in a car production line," *Advanced Robotics*, vol. 25, no. 3-4, pp. 473–489, 2011.
[18] D. Guo, F. Sun, B. Fang, C. Yang, and N. Xi, "Robotic grasping using visual and tactile sensing," *Information Sciences*, vol. 417, pp. 274–286, 2017.
[19] H. Liu, Y. Yu, F. Sun, and J. Gu, "Visual–tactile fusion for object recognition," *IEEE Tran. on Automation Science and Engineering*, vol. 14, no. 2, pp. 996–1008, 2017.
[20] O. Kroemer, C. H. Lampert, and J. Peters, "Learning dynamic tactile sensing with robust vision-based training," *IEEE Tran. on Robotics*, vol. 27, no. 3, pp. 545–557, 2011.
[21] P. Falco, S. Lu, A. Cirillo, C. Natale, S. Pirozzi, and D. Lee, "Cross-modal visuo-tactile object recognition using robotic active exploration," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, May 2017, pp. 5273–5280.
[22] W. Becari, L. Ruiz, B. G. Evaristo, and F. J. Ramirez-Fernandez, "Comparative analysis of classification algorithms on tactile sensors," in *Proc. of the Int. Sym. on Consumer Electronics*, 2016, pp. 1–2.
[23] E. Hyttinen, D. Kragic, and R. Detry, "Learning the tactile signatures of prototypical object parts for robust part-based grasping of novel objects," in *Proc. of the Int. Conf. on Robotics and Automation*, 2015, pp. 4927–4932.
[24] A. J. Spiers, M. V. Liarokapis, B. Calli, and A. M. Dollar, "Single-grasp object classification and feature extraction with simple robot hands and tactile sensors," *IEEE Tran. on Haptics*, vol. 9, no. 2, pp. 207–220, 2016.
[25] G. De Maria, C. Natale, and S. Pirozzi, "Force/tactile sensor for robotic applications," *Sensors and Actuators A: Physical*, vol. 175, pp. 60–72, 2012.
[26] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of the Conf. on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.
[27] C. Mitash, K. E. Bekris, and A. Boularias, "A self-supervised learning system for object detection using physics simulation and multi-view pose estimation," *arXiv preprint arXiv:1703.03347*, 2017.
[28] G. Georgakis, A. Mousavian, A. C. Berg, and J. Kosecka, "Synthesizing training data for object detection in indoor scenes," *arXiv preprint arXiv:1702.07836*, 2017.
[29] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Tran. on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142–158, 2016.
[30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
[31] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
[32] D. De Gregorio, F. Tombari, and L. Di Stefano, "Robotfusion: Grasping with a robotic manipulator via multi-view reconstruction," in *Computer Vision ECCV Workshops*. Springer, 2016, pp. 634–647.
[33] J. L. Bentley and T. A. Ottmann, "Algorithms for reporting and counting geometric intersections," *IEEE Tran. on computers*, no. 9, pp. 643–647, 1979.
[34] D. Gallup, J.-M. Frahm, P. Mordohai, and M. Pollefeys, "Variable baseline/resolution stereo," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
[35] R. Munoz-Salinas, "Aruco: a minimal library for augmented reality applications based on opencv," *Universidad de Crdoba*, 2012.
[36] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series." in *KDD workshop*, vol. 10, no. 16. Seattle, WA, 1994, pp. 359–370.

Daniele De Gregorio holds the position of Post-doctoral Researcher in the University of Bologna. His research interests include Robotics, Computer Vision and Deep Learning. He received the B.Cs. (2008) and the M.Cs. (2012) degrees from the University of L'Aquila, Italy and the PhD. degree (2018) from the University of Bologna, Italy in Software Engineering. Moreover, he has been working for over 10 years outside university as Software Consultant.

Riccardo Zanella is currently a PhD student in Biomedical, Electrical and System Engineering, with curriculum in Automatic Control and Operative Research, at the University of Bologna. He holds a B.Sc. degree in Information Engineering and a M.Sc. degree in Automation Engineering from the University of Padova. His doctoral work is on the development of intelligent robots for autonomous manipulation of deformable objects. He is especially interested in deep learning techniques applied to autonomous robots control based on vision and force sensing.

Gianluca Palli Gianluca Palli received the Laurea and the Ph.D. degrees in automation engineering from the University of Bologna, Italy, in 2003 and 2007, respectively. Currently he is Assistant Professor at the University of Bologna. His research interests include design and control of robotic hands, modeling and control of robots with variable stiffness joints, design of compliant structures and actuation systems for robotics applications and development of real-time systems for automatic control applications. He is the author or coauthor of more than 80 scientific papers presented at conferences or published in journals.

Salvatore Pirozzi currently holds the position of Associate Professor at University of Campania Luigi Vanvitelli. His research interests include modeling and control of smart actuators for active noise and vibration control, design and modelling of innovative sensors, in particular of tactile solutions, as well as interpretation and fusion of data acquired from the developed sensors. He published more than 70 international journal and conference papers. He currently serves as Associate Editor of the IEEE Transactions on Control Systems Technology.

Claudio Melchiorri Claudio Melchiorri (M92-SM03) received the Laurea degree in electrical engineering and the Ph.D. degree from the University of Bologna, Bologna, Italy, in 1985 and 1990, respectively. In 1998, he was an Adjunct Associate in engineering with the Department of Electrical Engineering, University of Florida, Gainesville. In 1990-1991 he was a Visiting Scientist with the Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge. Since 1985 he has been with the Department of Electrical, Electronic and Information Engineering, University of Bologna, where he is currently a Full Professor of robotics. His research interests include dexterous robotic manipulation, haptic interfaces, telemanipulation systems, advanced sensors, and nonlinear control. He is the author or coauthor of about 270 scientific papers presented at conferences or published in journals and of 13 books on digital control and robotics.