IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 22, 2021, accepted July 15, 2021, date of publication July 19, 2021, date of current version July 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3098472

Vision-Based Robotic Solution for Wire Insertion With an Assigned Label Orientation

PASQUALE CIRILLO, GIANLUCA LAUDANTE, AND SALVATORE PIROZZI
Dipartimento di Ingegneria, Università della Campania "Luigi Vanvitelli," 81031 Aversa, Italy

Corresponding author: Salvatore Pirozzi (salvatore.pirozzi@unicampania.it)

This work was supported by the European Commission within H2020 Robotic Technologies for the Manipulation of Complex Deformable Linear Objects (REMODEL) Project under Grant 870133.

ABSTRACT This paper tackles the problem of wire insertion in switchgear assembly according to the current regulations. In particular, the wire connections require that the wire label has to be oriented facing up in order to simplify and speed up testing and maintenance of the switchgear. The proposed approach exploits the a priori knowledge of the scenario with a calibrated RGB camera and a robotic arm to estimate both wire end pose and label position. The procedure combines several techniques (gradient base, trained classifier and stereo vision) to elaborate standard images in order to extract some wire features related to its shape and label. Specific frames are fixed according to estimated features and then used to correctly complete the task by using a robotic system. Experiments are reported to verify the effectiveness of the proposed approach.

INDEX TERMS Wire manipulation, image elaboration, switchgear assembly.

I. INTRODUCTION

The objects manipulation performed by using robotic arms and grippers, allowed a huge spread of robots in several production scenarios in which human work was widely adopted. However, when a high level of manipulation ability is required due to the object complexity, the human being is still forced to perform repetitive tasks that can harm his psycho-physical well-being. In order to overcome current limitations in automation of these tasks, the development of new approaches is fundamental. The success of proposed solutions mainly depends on the exploitation of actual methods together with appropriate improvements made on the basis of a priori knowledge of the scenario. The work described in this paper arises from an European project, termed REMODEL (Robotic tEchnologies for the Manipulation of cOmplex DeformablE Linear objects), aiming to increase the robots use in the manipulation of DLOs (Deformable Linear Objects), and in particular of wires, widely used in several sectors such as automotive and aerospace. In order to demonstrate the effectiveness of the REMODEL results, several industrial manufacturing use cases provided by the industrial partners will be developed. One of these is the switchgear wiring process, a task in which the precision in the manipulation of the DLOs is very important, and the human sensitivity is required to cope with unpredictable occlusions and assembly complexity due to the variety of layouts. According also to [1], [2] and references therein, in switchgear assembly the wiring is the most time-consuming subtask, by corresponding up to the 49% of the total assembly time. Only some preliminary operations (e.g., wire cutting, stripping, crimping of the ferule) are currently automatized. Instead, the remaining operations require a large use of paper documentation that implies long time and worker experience for its reading, interpretation and translation into the required assembly sequence. Additionally, it is not sufficient to execute an insertion with connection, but also the position of the wire label has to be taken into account. The wire connections have to be realized by maintaining the wire label facing up in order to allow the switchgear testing, maintenance and inspection. Figure 1 reports an example of an assembled switchgear, where the facing up position of the labels is clear. It results that, in most of the cases, these operations are almost completely manually executed. The partial automation of this task is hence an open problem, which can be tackled by separating the main task in subtasks: wire grasping; label position estimation; wire pose estimation; insertion by taking into account both wire pose and label position. In literature, there are few cases in which researchers partially tackled some subtasks from

The associate editor coordinating the review of this manuscript and approving it for publication was Saeid Nahavandi.

similar actual use cases. Some researchers tackle the problem of planning for manipulation of deformable objects [3], [4]. Almost always, the use of vision is the most used approach, due to the efficiency in data collection [5]–[7]. By considering only the grasping of deformable objects, the number of paper further reduces. In [8], a planning method for knotting/unknotting of DLOs is proposed. In [9], a vision-based control scheme to enable the robot to contact and then to manipulate the flexible PCB (Printed Circuit Board) is proposed. The problems related to the insertion task for an elastic beam into a hole are considered in [10]. In [11] the vision data are used for the deformable object classification, while in [12] the authors propose a vision based algorithm able to estimate the coordinates of a grasping point near an edge of the deformable object. Some researchers used vision data also for tracking of 2D deformable objects [13]. In other papers the wire manipulation has been tackled by combining vision and tactile data or by using only the tactile sensors. For example, in [14], [15] the authors propose a mechatronic approach for wire manipulation in some phases of the switchgear assembly by using only tactile data, but without taking into account the label position. In [16] the combination of both data has been used to optimize the insertion subtask, again without considering the label position.

The main contribution of this paper is the development of a vision-based robotic solution able to automatize the wire insertion (typical of switchgear assembly), by maintaining the wire label in a desired orientation. To reach this goal several intermediate contributions are presented. First, a wire extraction algorithm, suitably optimized in order to exploit the a priori knowledge of the scene, is proposed to overcome some limitations related to available standard techniques: this algorithm is based on a gradient approach suitably adapted to the wires. Secondly, a specific approach for the label position estimation is presented, by combining a gradient based technique with a suitably trained classifier. Finally, the wire terminal pose, considering also the label orientation, is obtained with an approach that combines a standard stereo vision technique with geometric considerations. The procedure is presented also in an algorithmic form to simplify implementation and reproducibility. The proposed solution is tested, by using a calibrated RGB camera and a robotic arm, in a simplified scenario with respect to the industrial one. The simplified scenario has been selected to demonstrate the proposed approach about the insertion with the label correctly oriented, without taking into account the problems specifically related to the insertion phase, that corresponds to the final subtask, already deeply tackled in [16]. The results of [16] together with the approach proposed in this paper can be used in future, to further improve the execution of the main assembly task.

II. TASK DESCRIPTION

Figure 2 reports a block scheme of the whole task with all subtasks highlighted. The scheme shows the interactions among the various algorithms detailed in the following,

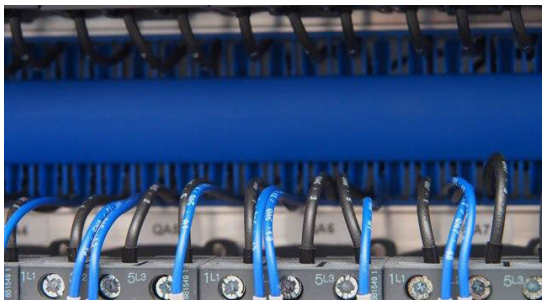


FIGURE 1. Typical switchgear wiring with the label facing up.

in order to simplify the readability of the paper. The task described in this paper consists in the insertion of an electric wire terminal in a hole that emulates the electromechanical component connector, by using a vision based approach. The objective is to achieve an insertion with a result similar to the one shown in Fig. 1, in which the labels printed on the wires are facing up. The label position and the wire terminal pose are estimated by analyzing RGB images of the wire itself. These information are then combined to perform a wire insertion with the label correctly oriented.

The whole task can be divided in five subtasks:

Wire grasping: the wire has to be grasped and positioned in front of the camera, in order to start the elaboration. The grasping point is supposed known, so this phase can be completed with a standard trajectory planning task, not described in this paper.

Wire extraction: an algorithm based on a gradient approach has been suitably designed for the detection of wire edges from a gray-scale image, by exploiting the a priori knowledge of the scene. The input of the algorithm is the whole image, while the output is a portion of the input image, i.e., the pixels corresponding to the wire. All other pixels are removed to reduce the computational load.

Label position estimation: in order to estimate the label position with respect to the camera, the robot arm rotates the wire and a suitably trained classifier is used to recognize the label. At the end of this phase, the output is a frame Σ_l with the z-axis outgoing from the label plane.

Wire terminal pose estimation: the wire terminal pose with respect to the camera frame is estimated by using a calibrated camera and a standard stereo vision approach. A frame Σ_s representing the estimated pose is positioned at the end of the wire. Then, by geometrically combining Σ_s and Σ_l, a third frame Σ_w is finally fixed, positioned at the wire end with an axis aligned with the label. This frame is the output of this subtask and it is used to achieve the insertion with the correct orientation of the label.

Wire insertion: the wire terminal is inserted in the hole by aligning Σ_w with the hole frame Σ_h, supposed known. A standard trajectory planning task is used to complete this phase.

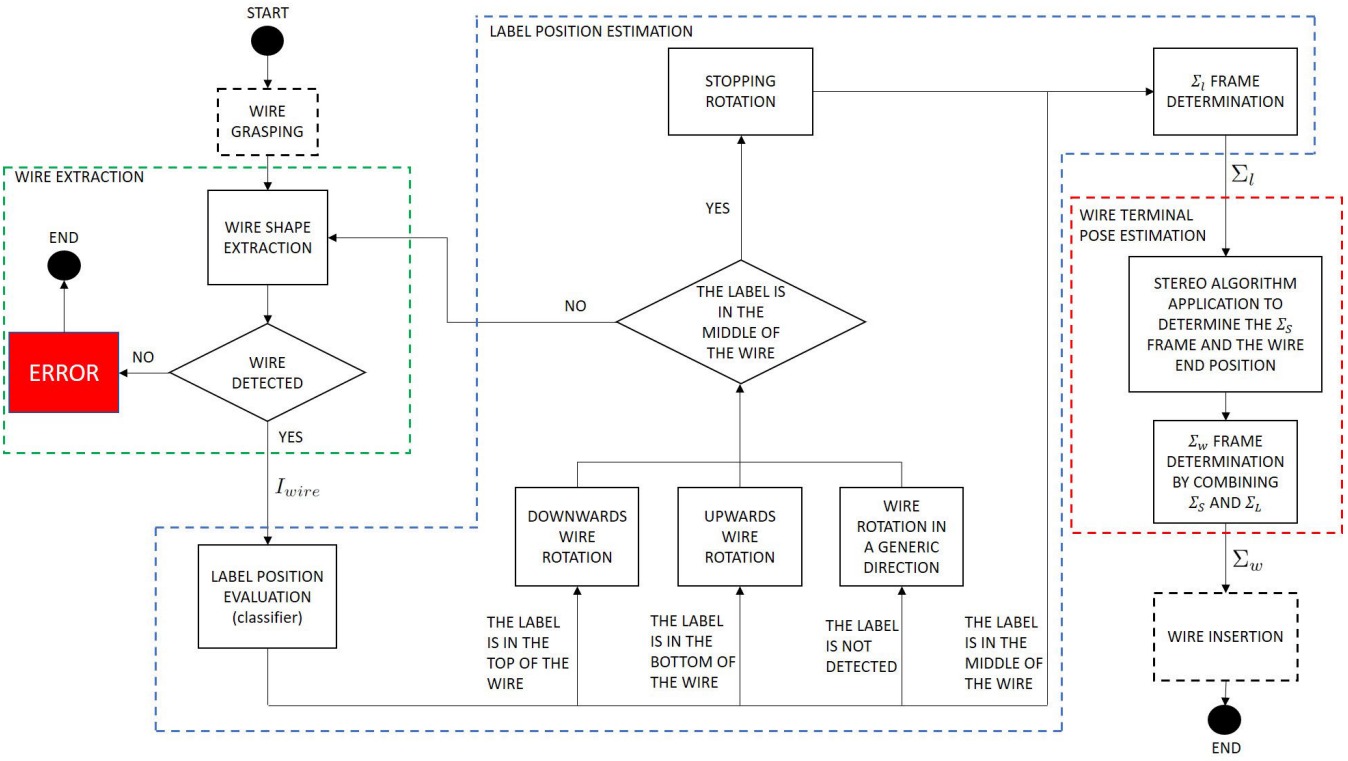
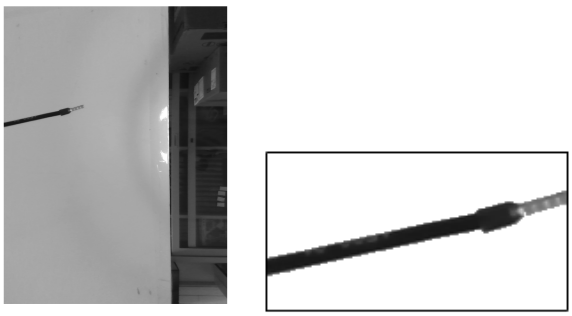


FIGURE 2. The whole scheme of the proposed solution, with all algorithms and their interactions.



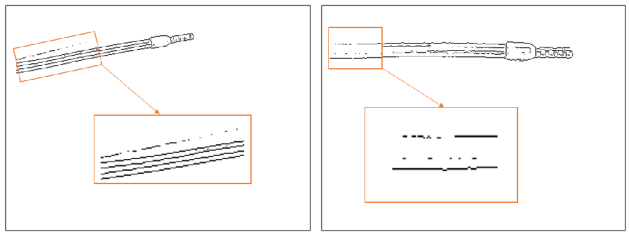
(a) (b)

FIGURE 3. Initial standard gray-scale image used for the elaboration (a) and output image of the wire extraction algorithm (b).

A. WIRE EXTRACTION

The objective of the wire extraction algorithm is to detect the wire edges and to extract only the wire pixels from the whole image. Figure 3a shows the typical gray-scale image (with values between 0 and 255) used as input of the proposed elaboration, defined as $I_g(i, j)$ with $i = 1, \ldots, M$ and $j = 1, \ldots, N$ representing the row and column indices of image pixels, respectively. An example of the output of this algorithm is shown in Fig. 3b.

Standard and well-known techniques based on a gradient approach, such as the Prewitt operator [17] or the Sobel operator [18], do not exploit any a priori knowledge of the scene, by always analyzing the entire image and by looking for gradient variations on the whole image and in any directions.



(a) (b)

FIGURE 4. Examples of incorrect wire edges extraction by using the Prewitt operator in particular light condition.

As a consequence, the number of pixels taken into account by standard approaches is always equal to the whole image pixels also in those cases (as the considered one) where only a small part of the image provides useful data. This makes the computational cost of the subsequent image manipulation unnecessarily high, since the pixels of interest are only the wire pixels. Moreover, the wire edges to extract have a predominant direction and are close together in a limited area, so it is useless to look for edges along other directions or other areas of the image. The standard approaches, since they analyze the whole figure in all directions and considering that the light conditions are not always optimal, often do not perform a correct wire edges detection, estimating edges not coherent with the actual ones (see Fig. 4). For these reasons, a suitable algorithm for the wire extraction is proposed in order to overcome encountered problems of standard approaches by exploiting the a priori knowledge of the scene.

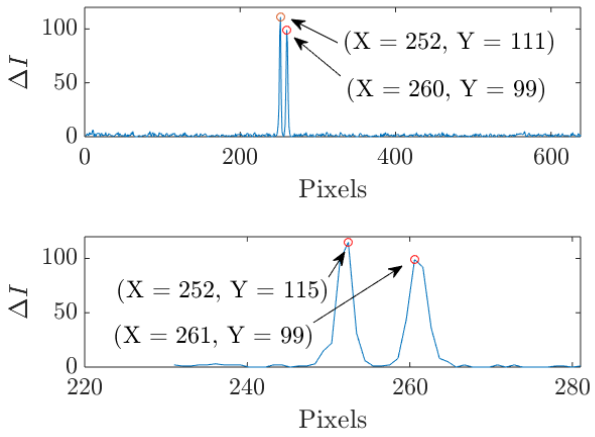


FIGURE 5. ΔI values for the first column of I_g (top) and for the second column of I_g (bottom).

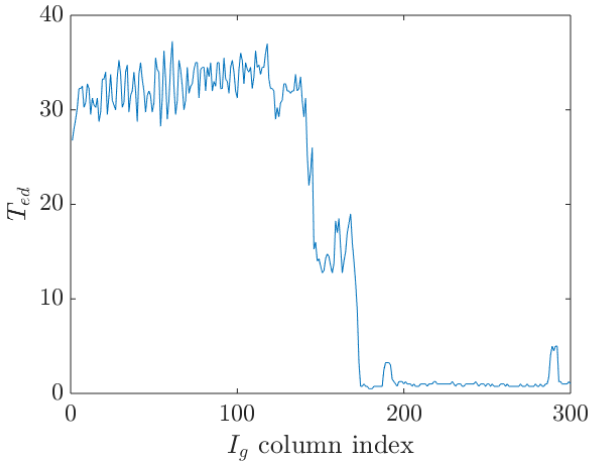


FIGURE 6. Threshold values computed for each column of the image in Fig. 3(a).

The developed algorithm exploits the gray-scale intensity variation to detect the wire edges. The variation is due to the color contrast between the wire and the background, as shown in the Fig. 3a. The wire grasping subtask guarantees that the cable is always present in the camera field of view, in the same side of the image (in this paper the left one) and with the same predominant direction (in this paper horizontal). The proposed wire detection algorithm uses this a priori knowledge of the scene to optimize the wire edges detection.

First of all, the algorithm computes the quantity $\Delta I(i,j) = |I_g(i,j) - I_g(i+\Delta n, j)|$ for the first column of the image ($j = 1$). The quantity ΔI presents two peaks in correspondence of wire edges, due to the different colour with respect to the background. Figure 5 (top) shows the $\Delta I(i,1)$ values for the first column of the image reported in Fig. 3a, with $\Delta n = 10$. The term Δn has to be chosen in order to maximize the difference between the two ΔI peaks and the other ΔI values due to image noise. The performance depends on the image vertical resolution: good results can be obtained with a Δn value ranging from $M/100$ to $M/30$. It is evident that the wire edges, related to these two peaks, are detectable by using a suitable threshold T_{ed} whose choice is fundamental for the correct edge detection. The use of a fixed value for all working conditions is not well performing and it may cause wrong estimation of the edges. This is mainly due to light conditions and possible shadows depending on the wire different characteristics (e.g., size, colour) and pose in the scene after the grasping subtask. To avoid this, an adaptive threshold $T_{ed}(j)$ is computed for each analyzed column of I_g, proportionally to the maximum value of $\Delta I(i,j)$

$$T_{ed}(j) = \max_i(\Delta I(i,j))/3 \tag{1}$$

By evaluating the pixels that verify the condition $\Delta I > T_{ed}$, the top edge ed_t and the bottom edge ed_b of the wire can be computed as the first pixel and the last pixel which verify the condition, respectively. Note that between the original pixel indices in I_g and the indices in ΔI, the offset due to Δn has to be compensated for the correct computation of ed_t and ed_b.

After the computation of wire edges for the first column, in order to determine the edges for the following columns, it is not necessary to analyze the entire columns, but it is sufficient to analyze only a portion of them, depending on the edges resulting from the previous column. This approach allows to reduce the algorithm computational load, by reducing the number of pixels to evaluate. In particular, indicating with $ed_t(j-1)$ and $ed_b(j-1)$ the pixels corresponding, respectively, to the top and bottom edges computed for the $(j-1)$-th column of $I_g(i,j)$, the portion of pixels to analyze for the j-th column is defined as

$$I_g([ed_t(j-1) - \Delta k, ed_b(j-1) + \Delta k], j) \tag{2}$$

The term Δk is a positive integer and has to be chosen to guarantee that the wire edges of the j-th column are included in the considered range of pixels, regardless of the wire inclination in the image. Its value can be fixed on the basis of image vertical resolution: a good choice ranges from $M/30$ and $M/10$. Hence, starting from the second column the threshold and the edges computation procedure is done considering only this reduced pixels range. Figure 5 (bottom) reports $\Delta I(i,2)$ computed for the pixels of the second column of I_g selected by using Eq. (2), with $\Delta k = 20$ and $\Delta n = 10$. It is evident that the number of analyzed pixels is much lower. The values used for Δk and Δn are always the same for all experimental data reported in the rest of this paper.

During the analysis of first columns it is fundamental to verify if the wire is correctly detected for the successful implementation of the whole task. This check can be done by evaluating if the estimated wire edges correctly correspond to a wire with an almost fixed diameter. From estimated edges, the wire diameter can be computed for each column as $w_d(j) = ed_b(j) - ed_t(j)$. The correct detection of the wire can be evaluated, starting from the second column ($j = 2$) and for a limited number N_c of subsequent columns (e.g., $N_c = 50$),

by using the following condition

$$w_d(j-1) - \frac{w_d(j-1)}{5} \leq w_d(j) \leq w_d(j-1) + \frac{w_d(j-1)}{5} \tag{3}$$

Equation (3) corresponds to evaluate if the computed wire diameter is almost constant and as a consequence if the estimated $ed_t(j)$ and $ed_b(j)$ are really the edges of a wire. If the condition in Eq. (3) is verified for the first N_c columns the wire detection can be considered correct, otherwise an error condition is raised.

By analyzing the threshold values computed with Eq. (1), reported in Fig. 6, it is evident that high values of $T_{ed}(j)$ are obtained only if the j-th column of the image includes the wire pixels. As a consequence, it is possible to detect the last column where wire edges are present (corresponding to the wire end) and to define a stopping criterion to not analyze the portion of the image without the wire, further optimizing the computational time. In particular, since the decrease of T_{ed} is right after the wire end, the stopping criterion can be obtained by evaluating if the j-th value of T_{ed} is lower than a fraction of its mean value computed on the previous $(j-1)$ values

$$T_{ed}(j) < mean(T_{ed}([1, j-1]))/3 \tag{4}$$

The last analyzed column before the stopping criterion is verified, defined as j_{max}, corresponds to the wire end. The output of the whole algorithm is the reduced size image $I_{wire}(i, j)$, with row and column indices ranging, respectively, in $[\min_j(ed_t(j)) - \Delta k, \max_j(ed_b(j)) + \Delta k]$ and $[1, j_{max}]$. In addition, for each column, the pixels between the computed edges remain equal to the original ones (from I_g), while the intensity value of the remaining pixels is set to 255 (white colour). The whole procedure is summarized in the pseudo-code reported in Algorithm 1.

The obtained result for the considered example image in Fig. 3a is the reduced image reported in Fig. 3b. The size of the output image I_{wire} is 90×171 pixels, while the size of the input image I_g is 640×480 pixels. This image reduction also implies that the computational cost of the subsequent image manipulation operations is reduced.

Algorithm 1 Wire Extraction

Input: $I_g(i, j) \quad i \in [1, M], j \in [1, N]$
Initialization: Δn, Δk ▷ Wire edges computation for the first column $j = 1$
$\Delta I(i, 1) = |I_g(i, 1) - I_g(i + \Delta n, 1)| \quad i \in [1; (M - \Delta n)]$
Compute $T_{ed}(1)$ ▷ from Eq. (1)
$edges = find(\Delta I(:, 1) > T_{ed}(1))$
$ed_t(1) \leftarrow edges(1) + (\Delta n - 1)/2$
$ed_b(1) \leftarrow edges(end) + (\Delta n - 1)/2$
$w_d(1) = ed_b(1) - ed_t(1)$ ▷ Wire edges computation from the second column $j > 1$
while $j < N$ **do**
  $\Delta I(i, j) = |I_g(i, j) - I_g(i + \Delta n, j)|$ ▷ from Eq.(2) $i \in [ed_t(j-1) - \Delta k, ed_b(j-1) + \Delta k]$
  Compute $T_{ed}(j)$ ▷ from Eq. (1)
  if $T_{ed}(j) < mean(T_{ed}([1, j-1]))/5$ **then**
    break ▷ stopping criterion from Eq. (4)
  else
    $edges = find(\Delta I(:, j) > T_{ed}(j))$
    $ed_t(j) \leftarrow edges(j) + (\Delta n - 1)/2$
    $ed_b(j) \leftarrow edges(end) + (\Delta n - 1)/2$
  end if
  if $j < N_c$ **then** ▷ Wire correct detection check
    $w_d(j) = ed_b(j) - ed_t(j)$
    Check *Eq.* (3)
    if *Eq.* (3) is not verified **then**
      break ▷ ERROR exit
    end if
  end if
  $j = j + 1$;
end while
$j_{max} = j$ ▷ Wire image extraction
$I_{wire}(i, j) = I_g(i, j)\ i \in [\min_j(ed_t(j)) - \Delta k, \max_j(ed_b(j)) + \Delta k], j \in [1, j_{max}]$
for $j = 1 : j_{max}$ **do**
  $I_{wire}(i < ed_t(j) \wedge i > ed_b(j), j) = 255$
end for
Output: I_{wire}
Additional outputs: $ed_t(j)$, $ed_b(j) \quad j \in [1, j_{max}]$

B. LABEL POSITION ESTIMATION

The objective of this subtask is to place the wire in front of the camera with the label in a desired position, i.e., in the middle of the wire. The proposed approach makes use of a suitably trained classifier in combination with a rotational movement of the wire, accomplished by means of the robotic system. The classifier is able to distinguish the following four situations:

1. label at the top of the wire (Fig. 7a);
2. label at the bottom of the wire (Fig. 7b);
3. label not present (Fig. 7c);
4. label in the middle of the wire (Fig. 7d);

These four detectable label positions have been selected as classifier output data. For each frame acquired by the camera, the proposed approach has a first pre-processing phase of the I_{wire} image (output of wire extraction algorithm explained before), in order to prepare inputs for the classifier. Then, on the basis of classifier output, the robotic system makes a suitable rotation until the situation 4 (i.e. label in the middle of the wire) is detected by the classifier. Starting from the worst case (label not present), a maximum rotation of 180^o is needed. The reduced number of elaborated pixels becomes fundamental, since it allows a fast elaboration of each frame.

The pre-processing phase exploits the arrays $ed_t(j)$ and $ed_b(j)$ to compute the pixels corresponding to the wire central axis $w_c(j)$ and the wire diameter $w_d(j)$ for each column of the image I_{wire}, i.e.,

$$w_c(j) = (ed_b(j) + ed_t(j))/2 \tag{5}$$

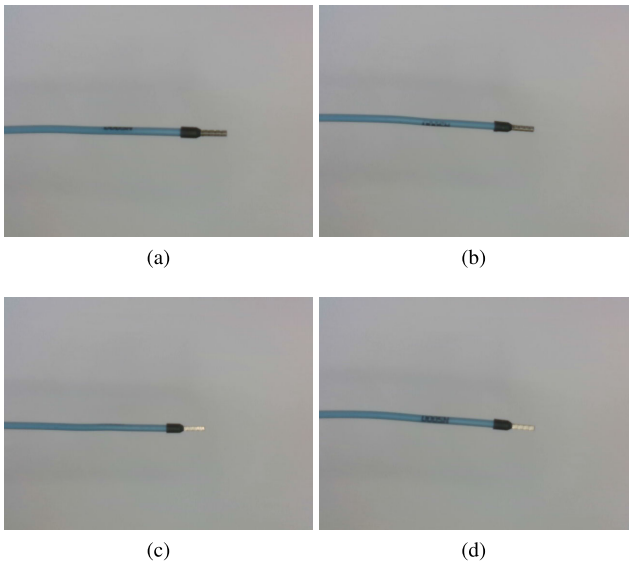


FIGURE 7. Example in which the label is in the top of the wire (a), example in which the label is in the bottom of the wire (b), example in which the label is not present (c) and example in which the label is in the middle of the wire (d).

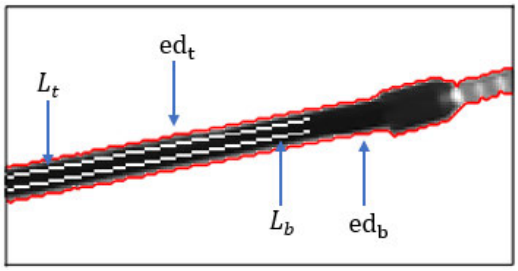


FIGURE 8. $L_t(j)$ and $L_b(j)$ arrays with respect to $ed_t(j)$ and $ed_b(j)$.

$$w_d(j) = ed_b(j) - ed_t(j) \tag{6}$$

To detect the position of the label, it is necessary to explore the wire along its longitudinal direction. To this aim, it is possible to define two arrays $L_t(j)$ and $L_b(j)$ parallel to the wire central axis $w_c(j)$. The former is positioned between $w_c(j)$ and $ed_t(j)$, while the latter between $w_c(j)$ and $ed_b(j)$,

$$L_t(j) = I_{wire}(w_c(j) - w_d(j)/r, j) \tag{7}$$
$$L_b(j) = I_{wire}(w_c(j) + w_d(j)/r, j) \tag{8}$$

where the parameter r defines the distance from the center of the wire and it can range in $[2, \infty)$, corresponding to a variation from the edges to the center. From a practical point of view, values ranging in $[3, 6]$ guarantee good performance. For all experiments in this paper $r = 4$. Since the terminal of the wire is not interesting in label estimation, on the contrary the corresponding pixels can cause problems to the classifier, it is not necessary to evaluate $L_t(j)$ and $L_b(j)$ for all I_{wire} columns. Therefore, the evaluation is limited to the 70% of I_{wire} columns, i.e., $j \in [1, 0.7j_{max}]$. Figure 8 shows an example of the $L_t(j)$ and $L_b(j)$ arrays highlighted with white pixels, together with $ed_t(j)$ and $ed_b(j)$ highlighted with red pixels. The information exploited to detect the label position is the chromatic difference between the label and the wire. Therefore, a similar method to the one used for the wire extraction algorithm, based on the gray-scale intensity variation, can be applied to $L_t(j)$ and $L_b(j)$ by defining the variations ΔL_t and ΔL_b

$$\Delta L_t = L_t([1, end - \Delta n]) - L_t([1 + \Delta n, end]) \tag{9}$$
$$\Delta L_b = L_b([1, end - \Delta n]) - L_b([1 + \Delta n, end]) \tag{10}$$

where *end* represents the last sample of the array. Figure 9 shows the ΔL_t and ΔL_b values corresponding to the different label positions reported in Fig. 7. It is straightforward to understand that their peak values can be used to estimate the label position with respect to the camera view. In fact:

- if the label is not present, the intensity variations are low for both $L_t(j)$ and $L_b(j)$, since they refer to two wire portions of the same color (e.g., see Fig. 9c);
- if the label is at the top (bottom) of the wire, the intensity variation is high only for $L_t(j)$ ($L_b(j)$), since the other one is not affected by the chromatic variation due to the label color (e.g., see Fig. 9a and Fig. 9b respectively);
- if the label is in the middle of the wire, the intensity variations are high for both $L_t(j)$ and $L_b(j)$, since both intersect the label pixels (e.g., see Fig. 9d).

Hence, the intensity variations ΔL_t and ΔL_b can be used as classifier input data. The classifier can be trained by using input and output data extracted from images as the ones reported in Fig. 7. For the experiments reported in this paper, data extracted from 16000 images have been used to train a neural network (Fig. 10) composed by a hidden layer with 10 neurons and an output layer that returns the value indicating the label position estimation among the four possible output cases, by elaborating the two ΔL_t and ΔL_b inputs. The Bayesian regularization back-propagation algorithm (*trainbr* MATLAB function) has been used to train the classifier by considering 100 iterations. The time requested for the training is about 5 min. It has been tested by using about 4000 new images (about 1000 for each of the four possible outputs), reporting a success rate close to 100%.

On the basis of classifier output, the robotic system executes a clockwise or a counterclockwise rotation of the wire until the label is in the middle of the wire. With the label centered, it is finally possible to fix a frame Σ_l with the z-axis outgoing from the image plane. It is worth to notice that, as detailed in the next section, only the z-axis is fundamental for the implementation of the final task. As a consequence, the x-axis can be fixed, for example, aligned with the wire central axis, and the y-axis is fixed in order to complete the orthogonal axes. The origin of the frame Σ_l is fixed at the end of the wire, as described in the next section.

The whole procedure for the label positioning is summarized in the pseudo-code reported in Algorithm 2.

C. WIRE TERMINAL POSE ESTIMATION

The objective is to fix a Σ_w frame at the wire end, which takes into account both the label position detected in the

previous section and the 3D shape of the wire end. To this aim, the frame Σ_l is geometrically combined with a frame Σ_s describing the wire pose with respect to the camera frame, defined as Σ_c.

1) POSITION ESTIMATION

In order to obtain 3D information from 2D images, a standard stereo vision approach has been used. But, from an implementation point of view, instead of using two parallel cameras, the proposed approach exploits the robotic system to acquire two parallel images with the same camera. These images are then elaborated with the stereo vision approach. Looking at Fig. 11, I_{g1} and I_{g2} represent two images captured by two parallel cameras at a distance b from each other. A generic point $P(X_p, Y_p, Z_p)$ in the 3D space corresponds to the pixel $p_1(u_1, v_1)$ on the first image and to $p_2(u_2, v_2)$ on the second one, where u and v are the pixel coordinates. Combining the data from the two images, it is possible to estimate the coordinate Z_p as

$$Z_p = \frac{f * b}{|p_1 - p_2|} \tag{11}$$

where f is the focal length of the camera. Then, by using one of the two images, it is possible to estimate also the remaining coordinates of P as

$$X_p = \frac{Z_p}{m_x f}(u - u_c) \tag{12}$$

$$Y_p = \frac{Z_p}{m_y f}(v - v_c) \tag{13}$$

where m_x and m_y are, respectively, the reciprocals of the width and height of a pixel on the camera plane, while $p_c(u_c, v_c)$ represents the image center. The intrinsic parameters f, m_x and m_y, if not available from datasheet, can be estimated by a standard calibration procedure of the camera. In the proposed approach the two images are obtained by moving the robot vertically. An example of typically obtained images I_{g1} and I_{g2} is reported in Fig. 12, where a vertical offset $b = 7$ cm has been chosen for the movement. In order to apply the Eq. (11), it is necessary to find the correspondence between the wire pixels in the two stereo images, i.e., the terms p_1 and p_2. To solve this problem, also known as correspondence problem, the wire extraction algorithm is applied to both I_{g1} and I_{g2} images, and the corresponding $w_{c1}(j), j \in [1, j_{max1}]$ and $w_{c2}(j), j \in [1, j_{max2}]$ arrays, i.e., the wire central axis, are computed according to Eq. (5). Figure 13 shows the elements of $w_{c1}(j)$ and $w_{c2}(j)$, computed for the images in Fig. 12, replaced with white pixels.

Algorithm 2 Label Positioning

```
while 1 do
    Execute: Wire extraction algorithm
    Compute w_c(j), w_d(j)            ▷ from Eq.(5),(6)
    Compute L_t(j), L_b(j)            ▷ from Eq.(7),(8)
    Compute ΔL_t(j), ΔL_b(j)          ▷ from Eq.(9),(10)
    switch classifier output do
        case 1
            rotate wire cw            ▷ Label @top
        case 2
            rotate wire ccw           ▷ Label @bottom
        case 3
            rotate wire cw            ▷ No label
        case 4
            break                     ▷ Label centered
end while
Output: Σ_l
```

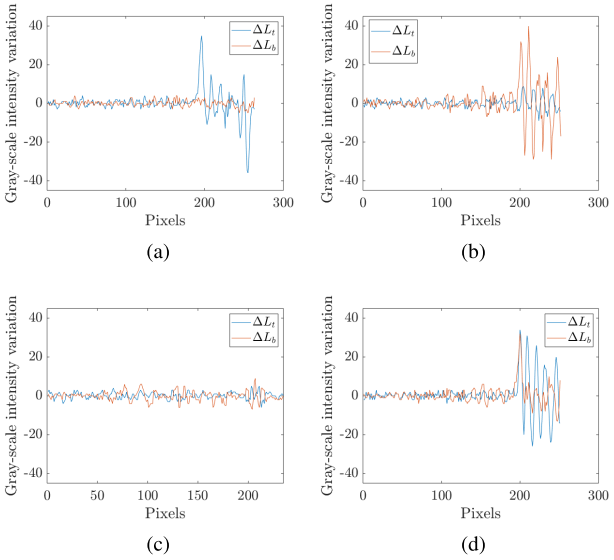


FIGURE 9. ΔL_t and ΔL_b when the label is in the top of the wire (a), ΔL_t and ΔL_b when the label is in the bottom of the wire (b), ΔL_t and ΔL_b when the label is not present (c) and ΔL_t and ΔL_b when the label is in the middle of the wire (d).

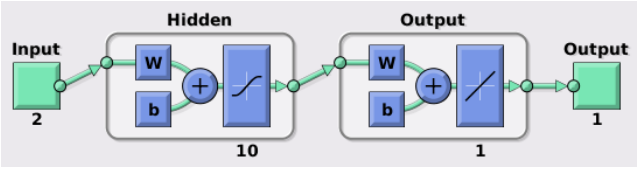


FIGURE 10. Classifier structure.

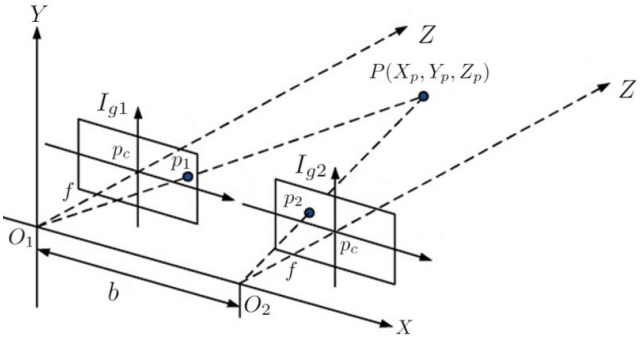
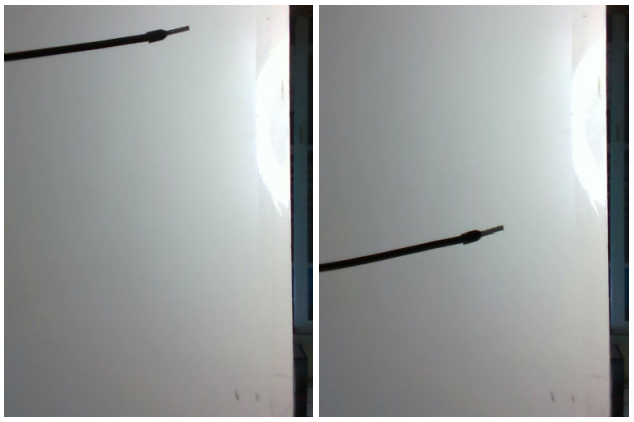
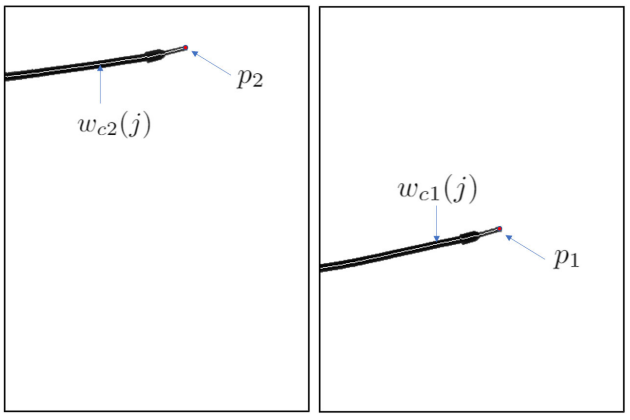


FIGURE 11. Stereo vision with parallel cameras.



(a) (b)

FIGURE 12. Stereo vision images I_{g1} (a) and I_{g2} (b) acquired to simulate two parallel cameras.



(a) (b)

FIGURE 13. Correspondence problem solved by determining the pixels of the wire central axis $w_c(j)$ for I_{g1} (a) and I_{g2} (b).

The knowledge of $w_{c1}(j)$ and $w_{c2}(j)$ allows to solve the correspondence problem, being pixels from two images corresponding to the same wire central axis. By applying the Eqs. (11)–(13) to the corresponding points in $w_{c1}(j)$ and $w_{c2}(j)$ arrays, the 3D coordinates for the whole wire central axis can be estimated. In particular, for the estimation of wire end position, the last elements of these arrays are necessary. The coordinates of these last elements are $p_1(w_{c1}(j_{max1}), j_{max1})$ and $p_2(w_{c2}(j_{max2}), j_{max2})$, and by applying the Eqs. (11)–(13) to p_1 and p_2, the wire end position P_w in the 3D scene can be determined.

2) ORIENTATION ESTIMATION

The wire terminal orientation with respect to the camera frame Σ_c is determined by exploiting the information about the orthogonal projection of the wire central axis on the two orthogonal planes in the 3D scene: the XY-plane (corresponding to the image plane IM) and the YZ-plane. Since all the coordinates of the wire central axis are known from the previous section, the coordinates of its projections are known as well. The projection on the XY-plane directly corresponds to w_{c2} in the IM-plane, while the projection on the YZ-plane

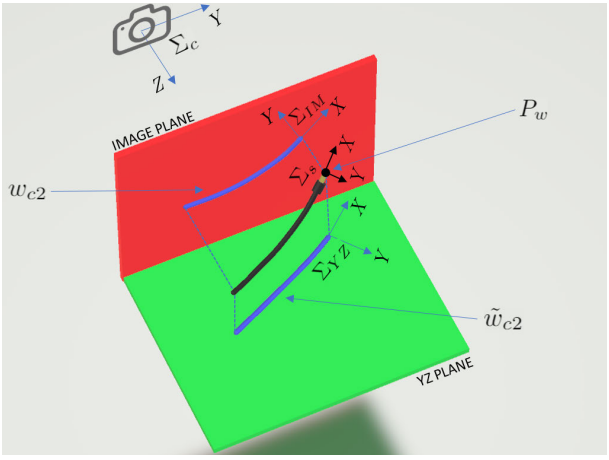


FIGURE 14. Geometric scheme for the orientation estimation.

is defined as $\tilde{w}_{c2}$. Figure 14 reports a geometric scheme of the wire orientation.

The frame Σ_s has the origin in the point P_w, i.e., the wire end point, and its orientation can be determined starting from the frame Σ_c and by applying a sequence of two rotations: $\mathbf{R}_Z^{IM}(\gamma)$ in the IM-plane (around the z-axis) and then $\mathbf{R}_X^{YZ}(\alpha)$ in the YZ-plane (around the x-axis)

$$\mathbf{R}_Z^{IM}(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{14}$$

$$\mathbf{R}_X^{YZ}(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \tag{15}$$

The single rotation matrices $\mathbf{R}_Z^{IM}(\gamma)$ and $\mathbf{R}_X^{YZ}(\alpha)$ describe, respectively, the orientation of the Σ_{IM} frame and Σ_{YZ} frame with respect to the camera frame, as reported in Fig. 14. The Σ_{YZ} and Σ_{IM} frames represent the projections of the wire terminal frame Σ_s on the two planes. The two angles γ and α can be determined from the angular coefficient of the tangent to the corresponding projection of the wire central axis, i.e., w_{c2} on IM-plane plane and $\tilde{w}_{c2}$ on YZ-plane, respectively. This computation can be done by interpolating each projection coordinates by means of a second order function ($L(u)$ for w_{c2} and $\tilde{L}(u)$ for $\tilde{w}_{c2}$) and by using the definition of angular coefficient, for both projections, separately

$$m = \left.\frac{dL(u)}{du}\right|_{u=P_w} \quad \Rightarrow \quad \gamma = \arctan(m) \tag{16}$$

$$\tilde{m} = \left.\frac{d\tilde{L}(u)}{du}\right|_{u=P_w} \quad \Rightarrow \quad \alpha = \arctan(\tilde{m}) \tag{17}$$

At this point, it is possible to compute the Σ_s frame by applying the obtained rotation matrices to the camera frame Σ_c. The resulting Σ_s frame has the origin in P_w and the $x-$axis, outgoing from the wire end, aligned with the tangent to the wire central axis in P_w.

Finally, the two frames Σ_s and Σ_l, describing respectively the pose of the wire terminal and the label position with

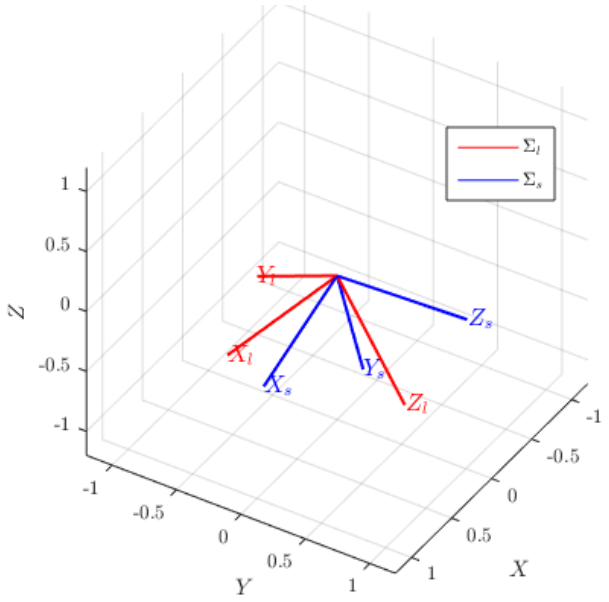


FIGURE 15. Σ_s and Σ_l frames.

respect to the camera frame, have to be appropriately combined in order to estimate the Σ_w frame. Figure 15 reports the two frames Σ_s and Σ_l applied in the same origin, to show that they are not related by a simple rotation. The looked for Σ_w must have the $x-$axis aligned with the $x-$axis of Σ_s and the $z-$axis on a plane orthogonal to $x-$axis and containing the $z-$axis of the label frame Σ_l. Figure 16 reports a graphical explanation for the determination of Σ_w. The Σ_s frame is reported in blue and the Σ_l in red. The green plane is orthogonal to x-axis of Σ_s, which corresponds to the x-axis for Σ_w and is highlighted in black in Fig. 16b. The yellow plane is obtained as a plane orthogonal to the green one and on which the $z-$axis of Σ_l lies. The z-axis of Σ_w frame, instead, is aligned along the intersection between the yellow and green planes (in black in Fig. 16b). The y-axis is fixed in order to complete the orthogonal axes and the origin is fixed in the wire end point P_w. The obtained Σ_w frame provides two fundamental information for the insertion task:

1) the wire terminal orientation with respect to the camera frame, provided by the x-axis;
2) a reference to the label position, provided by the z-axis.

The procedure for the wire end pose estimation is summarized in the pseudo-code reported in Algorithm 3.

D. WIRE INSERTION

The Σ_w frame is then used within a standard trajectory planning task in order to insert the wire into a mechanical hole with the desired orientation of the label. The mechanical component connectors have known positions, since CAD models of the whole switchgear and of electromechanical devices are available [19]. Hence, the insertion hole can be identified with a known frame Σ_h and the desired insertion (with the label oriented along a desired axis) can be completed aligning the Σ_w frame with Σ_h. Figure 17 reports the mechanical hole and

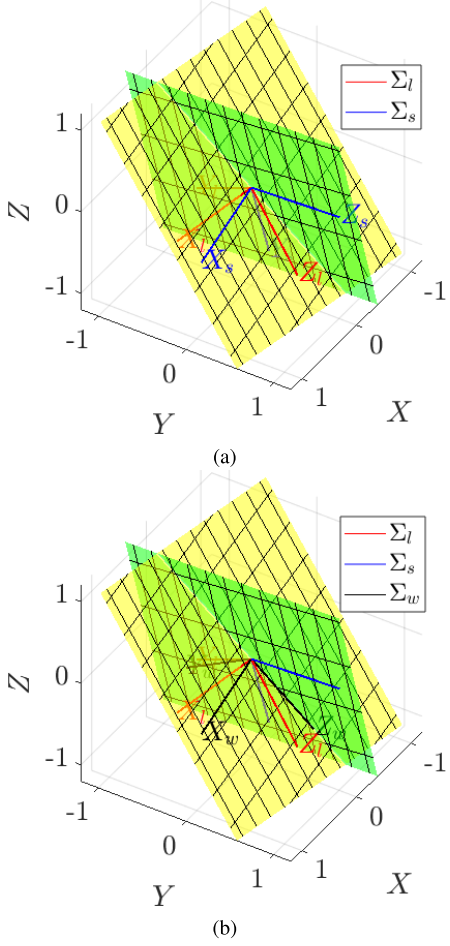


FIGURE 16. Geometrical combination between Σ_s and Σ_l to obtain Σ_w: plane combination (a) and frame definition (b).

corresponding frames used in this paper. The Σ_h frame has the x-axis pointing into the hole and aligned with its axis and the z-axis outgoing from the image. By considering the latter as the desired orientation of the label, the insertion can be correctly completed by aligning the Σ_w and Σ_h frames.

III. EXPERIMENTS

In this section experimental results are presented in order to demonstrate the effectiveness of the proposed procedure.

A. EXPERIMENTAL SETUP

Figure 18 shows the experimental setup and the frames considered to complete the whole task. The camera used is an Intel D435, but it is important to emphasize that the only information exploited in the described algorithm are the RGB images, so any RGB camera can be used. The gripper used for the wire manipulation is a SCHUNK WSG32, but any parallel gripper can perform this task. The robot used to test the algorithm is the Universal Robot UR5e, however it can be replaced by any robot that ensures the necessary degrees of freedom to perform all needed trajectories. Therefore, the developed algorithm is completely independent from the hardware. The only requested feature for the correct

Algorithm 3 Wire Terminal Pose Estimation

Input: f, m_x, m_y, u_c, v_c ▷ camera parameters
Acquisition of I_{g1} image
Apply a b displacement by using the robot
Acquisition of I_{g2} image
Execute the wire extraction algorithm on I_{g1} and I_{g2}
Compute w_{c1} and w_{c2} ▷ from Eq.(5)
Extract p_1 and p_2 from w_{c1} and w_{c2}
Compute 3D coordinates ▷ from Eqs. (11)–(13)
Extract from 3D coordinates P_w
Extract from 3D coordinates w_{c2} and $\tilde{w}_{c2}$
Compute from w_{c2} and $\tilde{w}_{c2}$ the functions $L(u)$ and $\tilde{L}(u)$
Compute γ and α ▷ from Eqs. (16)–(17)
Compute $\mathbf{R}_Z^{IM}(\gamma)$ and $\mathbf{R}_X^{YZ}(\alpha)$ ▷ from Eqs. (14)–(15)
Compute Σ_s
Compute from Σ_s and Σ_l planes in Fig. 16
Compute the output: Σ_w

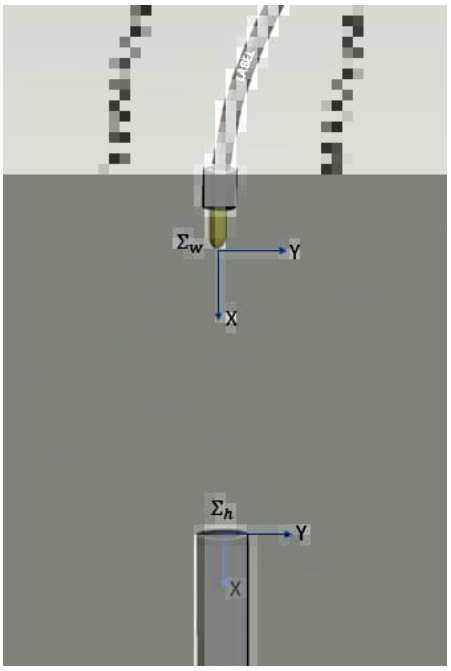


FIGURE 17. Alignment between the x-axis of the hole frame Σ_h and the x-axis of the wire frame Σ_w, while the label is aligned with the z-axis outgoing from the image.

implementation of the task is the white panel on the left in Fig. 18, placed in front of the camera. It is important for the wire extraction algorithm, in which the color contrast between the background and the wire is exploited to detect the wire edges.

B. EXPERIMENT DESCRIPTION

The proposed procedure has been tested with wires prepared by using the KOMAX Zeta 630 machine. The machine is able to prepare, with the same type of label, wires whose dimensions ranging from AWG (American Wire Gauge) 10 to AWG 24 (corresponding to an external diameter ranging from about

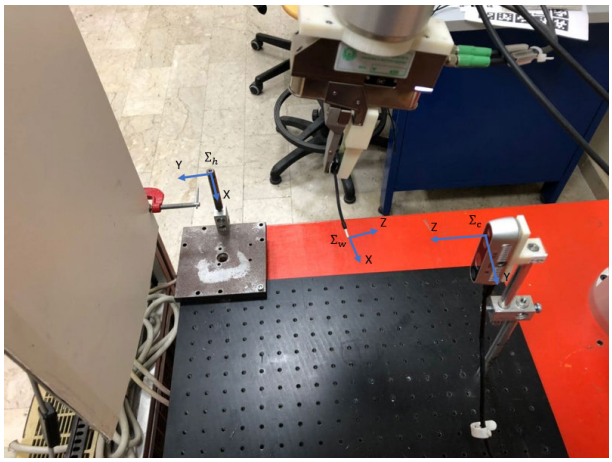


FIGURE 18. Experimental setup.

4.5 mm to about 2 mm). However, the experiments considered in this paper are mainly concentrated on wires with diameter among 2.5 mm to 3.5 mm, being the most used in switchgear assembly. For these used wires a standard $M6$ (corresponding to a diameter of about 5.5 mm) hole has been considered as target position. This choice has been made to simulate the switchgear electric components, which typically present connection holes with dimensions double with respect to wire diameter that have to receipt. Both blue and black wires have been used as it is visible along the paper figures. Note that few tests have been conducted on wires with diameter smaller than 2.5 mm, by highlighting some problems mainly related to the camera resolution, too low for these wires. Instead the task has been correctly executed for some tests carried out on wires with diameter higher than 3.5 mm, since the label is always well visible for larger wires.

The whole procedure has been tested about 90 times with the selected wires starting from different initial label positions and different wire deformations, i.e., the wire terminal pose with respect to the camera frame is different for each test case. Similar results have been obtained in all considered tests, by reaching a successful execution of about 87.7%, independently from wire diameter (ranging from 2.5 mm to 3.5 mm) and colour (black or blue).

Here a specific experiment for a black wire with a diameter of 3 mm is detailed. It starts from the worst case in terms of label position (it is not visible at beginning) and by explaining all steps executed to complete a wire insertion task. A video attached to the paper shows the whole execution. The sequence of the five subtasks described above has been executed. A standard trajectory planning task, here not detailed, has been implemented for the wire initial grasping. This subtask ends with the positioning of the wire in front of the camera as shown in Fig. 18. From this initial configuration, the Algorithm 1 and the Algorithm 2 are used to analyze the acquired RGB images and to provide the rotation commands to the robot until the label becomes centered. In the reported test, the algorithm detects that the label is not present (see Fig. 9a) at the beginning, so a clockwise rotation

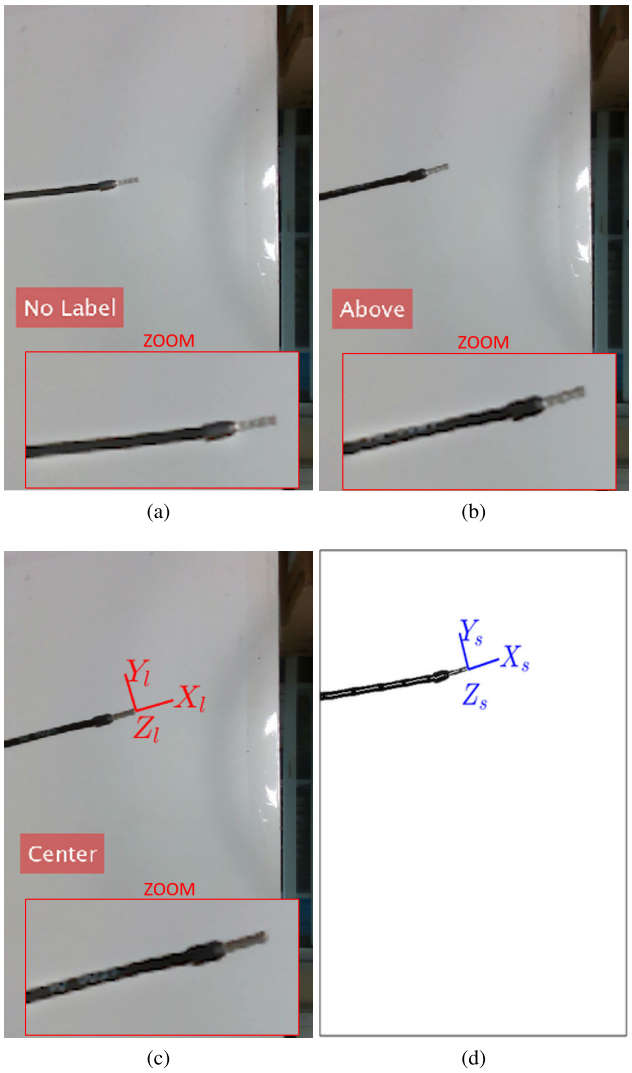


FIGURE 19. Label position estimation algorithm output: no label case (a) at beginning, detection of label in the upper part of the wire (b) during rotation, centered label corresponding to the end of rotation with estimated Σ_l (c). Estimated Σ_s by using the stereo vision approach with Algorithm 3 (d).

motion starts. During the rotation, the label appears in the upper part of the wire and the situation is correctly detected by the classifier (see Fig. 19b). The rotation continues until the label appears centered (see Fig. 19c). At the end of the rotation, the frame Σ_l described in the previous section is determined.

The following subtask concerns the determination of Σ_s frame, computed by using the stereo vision. The wire is vertically shifted and the second image is acquired in order to apply the Algorithm 3 to determine Σ_s, as reported in Fig. 19d. At this point, starting from Σ_s and Σ_l frames, the Σ_w frame is computed. Finally, a standard trajectory planning is implemented in order to align the estimated Σ_w frame with the Σ_h hole frame as shown in Fig. 20. The orientation of the z-axis of Σ_w with respect Σ_h provides the desired orientation of the label: in this case it is aligned with the z-axis of Σ_h, outgoing from the image.

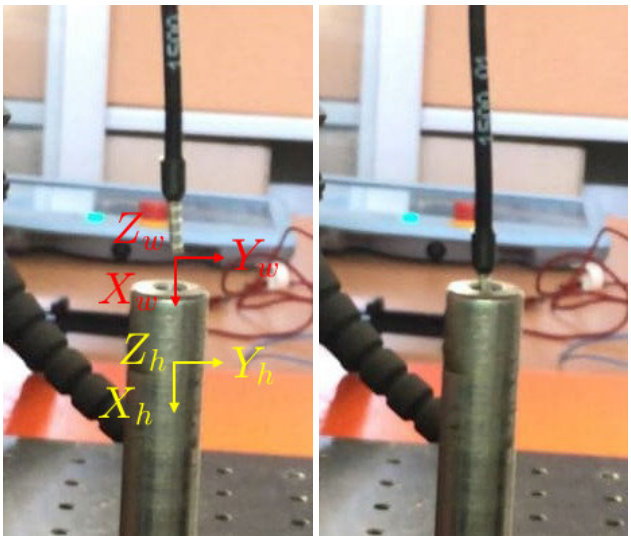


FIGURE 20. Wire insertion: alignment between Σ_w and Σ_h (a) and final insertion with desired orientation of the label (b).

IV. CONCLUSION AND FUTURE WORKS

This paper presented a suitably designed approach to tackle the problem of the estimation of the wire label position in switchgear assembly, according to current regulations. The label estimation has been combined with the wire end pose estimation in order to determine a wire end frame with the z-axis that takes into account the label position. By using this frame, an insertion phase has been implemented in order to demonstrate that it is possible to complete the task with the label oriented along a specified direction. Experiments demonstrated the effectiveness of the proposed approach in a simplified scenario, where the whole task has been completed by using only the vision. The presented results will be combined with previous results obtained in [16] in order to improve the performance during the insertion subtask.

REFERENCES

[1] F. Hefner, S. Schmidbauer, and J. Franke, "Vision-based adjusting of a digital model to real-world conditions for wire insertion tasks," *Procedia CIRP*, vol. 97, pp. 342–347, Jan. 2021.
[2] F. Hefner, S. Schmidbauer, and J. Franke, "Pose error correction of a robot end-effector using a 3D visual sensor for control cabinet wiring," *Procedia CIRP*, vol. 93, pp. 1133–1138, Jan. 2020.
[3] P. Jiménez, "Survey on model-based manipulation planning of deformable objects," *Robot. Comput.-Integr. Manuf.*, vol. 28, no. 2, pp. 154–163, 2012.
[4] T. Hermansson, R. Bohlin, J. S. Carlson, and R. Söderberg, "Automatic assembly path planning for wiring harness installations," *J. Manuf. Syst.*, vol. 32, no. 3, pp. 417–422, 2013.
[5] H. Nakagaki, K. Kitagaki, T. Ogasawara, and H. Tsukune, "Study of deformation and insertion tasks of a flexible wire," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 3, Apr. 1997, pp. 2397–2402.
[6] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *Int. J. Robot. Res.*, vol. 27, no. 2, pp. 157–173, 2008.
[7] M. Popović, D. Kraft, L. Bodenhagen, E. Başeski, N. Pugeault, D. Kragic, T. Asfour, and N. Krüger, "A strategy for grasping unknown objects based on co-planarity and colour information," *Robot. Auton. Syst.*, vol. 58, no. 5, pp. 551–565, May 2010.
[8] H. Wakamatsu, E. Arai, and S. Hirai, "Knotting/unknotting manipulation of deformable linear objects," *Int. J. Robot. Res.*, vol. 25, no. 4, pp. 371–395, Apr. 2006.

[9] X. Li, X. Su, and Y.-H. Liu, "Vision-based robotic manipulation of flexible PCBs," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2739–2749, Dec. 2018.
[10] H. Nakagaki, K. Kitagaki, and H. Tsukune, "Study of insertion task of a flexible beam into a hole," in *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 1, Apr. 1995, pp. 330–335.
[11] L. Martínez, J. Ruiz-del-Solar, L. Sun, J. P. Siebert, and G. Aragon-Camarasa, "Continuous perception for deformable objects understanding," *Robot. Auton. Syst.*, vol. 118, pp. 220–230, Aug. 2019.
[12] A. Sardelis, N.-C. Zacharaki, Z. Arkouli, D. Andronas, G. Michalos, S. Makris, and G. Papanikolopoulos, "2-stage vision system for robotic handling of flexible objects," *Procedia CIRP*, vol. 97, pp. 491–496, Jan. 2021.
[13] A. Petit, V. Lippiello, G. A. Fontanelli, and B. Siciliano, "Tracking elastic deformable objects with an RGB-D sensor for a pizza chef robot," *Robot. Auton. Syst.*, vol. 88, pp. 187–201, Feb. 2017.
[14] S. Pirozzi and C. Natale, "Tactile-based manipulation of wires for switchgear assembly," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2650–2661, Dec. 2018.
[15] G. Palli and S. Pirozzi, "A tactile-based wire manipulation system for manufacturing applications," *Robotics*, vol. 8, no. 2, p. 46, Jun. 2019.
[16] D. De Gregorio, R. Zanella, G. Palli, S. Pirozzi, and C. Melchiorri, "Integration of robotic vision and tactile sensing for wire-terminal insertion tasks," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 2, pp. 585–598, Apr. 2019.
[17] J. M. S. Prewitt, "Object enhancement and extraction," *Picture Process. Psychopictorics*, pp. 75–149, 1970.
[18] I. Sobel and G. Feldman, "A 3×3 isotropic gradient operator for image processing," *Pattern Classification Scene Anal.*, pp. 271–272, 1973.
[19] G. Palli, S. Pirozzi, M. Indovini, D. De Gregorio, R. Zanella, and C. Melchiorri, "Automatized switchgear wiring: An outline of the wires experiment results," in *Advances in Robotics Research: From Lab to Market* (Springer Tracts in Advanced Robotics), vol. 132. Cham, Switzerland: Springer, 2020, pp. 107–123.

PASQUALE CIRILLO received the master's degree *(cum laude)* in computer engineering from the University of Campania "Luigi Vanvitelli," Italy. He is currently a Research Fellow with the University of Campania "Luigi Vanvitelli." His research interests include development of suitable algorithms for the robotic automation of industrial processes, image segmentation and elaboration, and robot programming.

GIANLUCA LAUDANTE received the master's degree *(cum laude)* in computer engineering from the University of Campania "Luigi Vanvitelli," Italy, where he is currently pursuing the Ph.D. degree. His research interests include sensor development and characterization, data interpretation, robot control, and machine learning.

SALVATORE PIROZZI is currently an Associate Professor with the University of Campania "Luigi Vanvitelli," Italy. He has authored or coauthored more than 80 international journal and conference papers. His research interests include modeling and control of smart actuators for active noise and vibration control, design and modeling of innovative sensors, and interpretation and fusion of data acquired from the developed sensors. He is currently an Associate Editor of the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.