Automatic Color Inspection for Colored Wires in Electric Cables

Stefano Ghidoni, *Member, IEEE,* Matteo Finotto, and Emanuele Menegatti, *Member, IEEE*

***Abstract*—In this paper, an automatic optical inspection system for checking the sequence of colored wires in electric cable is presented. The system is able to inspect cables with flat connectors differing in the type and number of wires. This variability is managed in an automatic way by means of a self-learning subsystem and does not require manual input from the operator or loading new data to the machine. The system is coupled to a connector crimping machine and once the model of a correct cable is learned, it can automatically inspect each cable assembled by the machine. The main contributions of this paper are: (i) the self-learning system; (ii) a robust segmentation algorithm for extracting wires from images even if they are strongly bent and partially overlapped; (iii) a color recognition algorithm able to cope with highlights and different finishing of the wire insulation. We report the system evaluation over a period of several months during the actual production of large batches of different cables; tests demonstrated a high level of accuracy and the absence of false negatives, which is a key point in order to guarantee defect-free productions.**

Note to Practitioners

This work is motivated by the need of performing an accurate quality control on cable production: an automated inspection method is necessary for effectively assuring a quality check on 100% of the produced parts. The vision system exploits a compact acquisition hardware enabling the system to be easily integrated in existing cable crimping machines. The software system is composed by two main modules: the first one localizes the wires, while the second performs color measurements. The paper explains how it is possible to segment wires also when they are bent in many different ways; moreover, a reliable method for identifying colors, which is robust to wire markings and highlights, is described. The proposed system is able to automatically adjust tolerances in color measurement depending on the colors of the wires to be checked. This represents a strong point of the system, and it can be applied also in other contexts where color analysis on noisy data needs to be performed.

***Index Terms*—Visual inspection, wire color measurement, wire color sequence, wire detection, cable crimping.**

I. Introduction

Automatic visual inspection represents a strong advance in the field of quality control for industry, and is exploited since decades [1], [2], [3]. Computer vision algorithms applied to the industrial production environment can be considered as a mean for achieving better quality and lower costs in production. Automatic inspection systems can be effectively employed for performing complex quality controls and, most importantly, to check 100% of produced items instead of few samples along the production batch. For instance, this is extremely important in semiconductor production, where visual inspection is widely used [4], [5]. Since automatic visual inspection is not invasive, does not involve dangerous processes, and is essentially clean, it can be successfully employed in any industrial process, ranging from heavy industry [6] to food processing [7]. A wide range of sensors can be exploited in automatic visual inspection, like near infrared cameras [8], far infrared cameras [9], [10], X-ray cameras [11] or even ultrasound imaging [12].

In this paper, a system for inspecting assembly of electric connectors is presented. The system was integrated in an existing cable crimping machine, which could not be modified, as shown in figure 1. The crimping machine loads an empty connector and waits for a human operator to insert, in the connector holes, one wire after the other. Each time a wire is inserted, it is crimped to the connector. When a connector is completed, another one is loaded, without any stop or gap between the connectors.

This inspection system is not checking the crimping process. For the sake of this application, the crimping process can be considered error-free, since the quality of the connectors, wires and the machine itself is so high to ensure very limited errors, which, in case, can be detected by the crimping system.

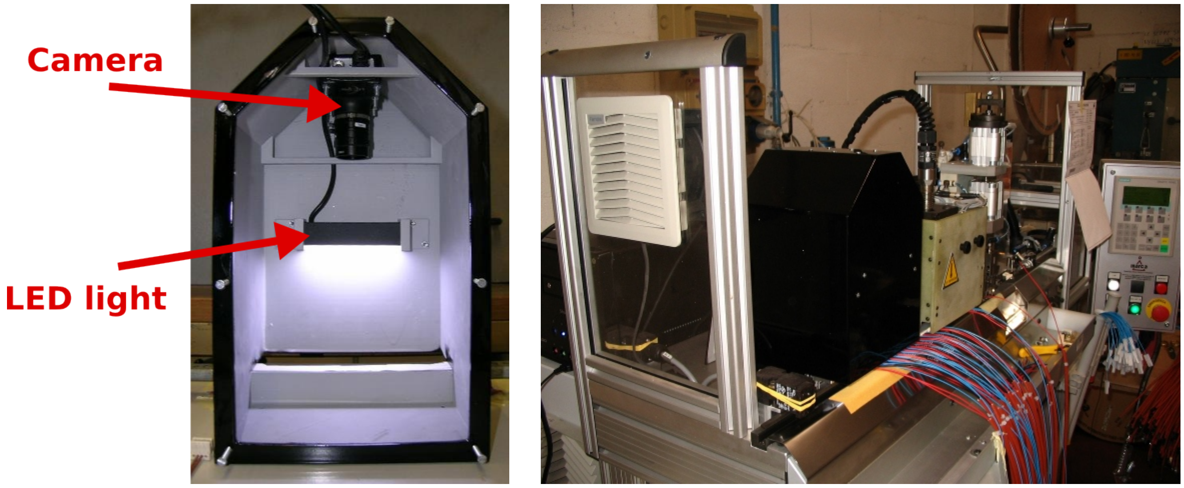
The whole process can be considered affected only by one source of error, namely the human operator inserting the wires in the wrong color order. Since wires have to satisfy a specific color coding, which has an electric meaning, crimping wires in the wrong order can lead to damages or malfunctions in the final products, and should be carefully avoided.

In order to verify if a cable has been correctly assembled, the system needs to be able to reliably locate the wires of each single cable and accurately identify their color. Even if the image acquisition is performed under controlled diffused light, highlights on PVC insulation and discontinuities in the wire color caused by cable markings is unavoidable. To cope with this, a reliable segmentation algorithm has been developed, based on the careful analysis of the scene framed by the camera. High accuracy in color recognition has been achieved by means of the proposed color extraction algorithm, which is able to robustly assess the color and detect the type of insulation of each wire.

The method presented in this paper is able to recover the wire color even under severe noise affecting more than 50% of the pixels belonging to the wire. This result would have been impossible to achieve using color histogram-based algorithms [25] or any other general purpose color segmentation technique, which does not rely on a model of the process being inspected and of the noise factors.

S. Ghidoni and E. Menegatti are with the Department of Information Engineering, University of Padova, Padova, Italy (email: {ghidoni,emg}@dei.unipd.it).

M. Finotto is with IT+Robotics srl, a spin-off company of the University of Padova, Padova, Italy (email: matteo.finotto@it-robotics.it)



(a) (b)

Fig. 1. The box containing the acquisition and lighting system (a), and the crimping machine during the crimping process after the installation of the visual inspection system (b).

Another important feature of the system is the verification of the correct sequencing of the wires. As we said, since the connectors appear without discontinuities in the images, the wires should be identified as belonging to different cables without knowing where the connectors are located. The solution proposed here is to analyze the wire sequence in a continuous production rather then considering cables singularly observed one after the other.

On this machine quality control was formerly performed on samples randomly extracted from produced items. This technique cannot guarantee 100% error-free production. Additionally, in this machine errors are due to sporadic human negligence, thus checks based on random sampling are even less effective, because errors cannot be modeled with an error distribution slowly deviating from the correct values, but rather as a Dirac delta distribution. From the above discussion, it is clear that a check of all produced items is desirable in order to provide a 100% error free production. A system capable of effectively performing visual inspection for this production must meet strong criteria, namely: (i) provide a method for checking all different types of produced items; (ii) do not slow down the production, i.e. the analysis should take shorter time than what is needed for producing a connector; (iii) represent a cost-effective solution. Our system successfully met all the three criteria.

The paper is organized as follows: in Section II related works are revised and relevant algorithms and systems are analyzed to understand similarities and differences with our system. In Section III the proposed system is described, including hardware setup, design principles and user interface. In Section IV the image analysis algorithms proposed for solving the discussed visual inspection problem are described in detail; this includes wire identification, color indexing and connector sequencing. In Section V experiments run for assessing the system performance in terms of detection accuracy, true positive rate and computational load are discussed, and their outcomes are reported. Finally, in Section VI a summary of the work done is reported, together with some final remarks.

II. Related work

Visual inspection for industrial production is a very active field, that spans across several sectors, ranging from food production [13], [7] and medical production [14], to fabric production, exploiting also quite complex computer vision techniques derived from other fields, as it is the case of [15]. Quality check systems based on visual inspection can be very sophisticated, and capable of interfacing with CAD models [16].

The exploitation of computer vision techniques in the industrial environment can lead to very successful results, but requires special care in the selection of the hardware components and setup [17]. Another strong constraint that visual inspection systems must meet is represented by the capability of being real-time, i.e. they should be able to check the production without affecting the production process speed [18]. From this point of view, visual inspection systems are similar to robotic vision systems, that must be able to extract data from the environment in real time so that the robot can take proper actions in time.

Several works in the literature have focused on metal parts [19] and electric connections: in [20], a system for inspecting metal connectors is presented, particularly focused on checking dimensional constraints; a similar type of inspection is also described in [21]. A number of works face the topic of wire bonding also at the microscopic scale, like it happens in the field of integrated circuit production [22].

Analysis of color information is a widely explored field in computer vision for any kind of applications, including visual inspection [23], [24]. Color indexing is often tackled by means of histograms [25], that are a convenient way for managing color information and creating clusters of similar colors. More sophisticated techniques for handling color information also exist in the literature: in [26] moments of color distributions are considered, while color signature based on bag of colors are presented in [27]; color itself can be described in a number of different color spaces, that can ease the task of discriminating one from another [28]. The techniques mentioned above were mainly developed to work on real-world scenes, as it often happens in robotics and computer vision applications like video surveillance or object recognition. They can deal with objects that have non-uniform color, and with scenes that undergo illumination changes, but do not aim at an accurate color measurement. On the contrary, such effects do not exist in an industrial context, because it is often possible to control illumination and the imaging process; however, in industrial visual inspection a much higher accuracy in color measurement is needed. This is the case of the work presented here: illumination is obtained by means of a LED illuminator, and external light is shielded. This makes our case rather different, since we have a strong knowledge of the phenomenon that is observed e.g. the cylindrical shape of the wire that causes a gradient that is repeated on all wires. The histogram-based approach is not suitable in our case, because it loses the spatial information: all colors are organized based on their values, while we need to discriminate between them based on geometrical considerations, e.g. the peculiar shape of the wires. It should be noted that the specific approach developed in our case relies on the *a priori* knowledge of the problem, while histogram-based methods are employed when such knowledge is not available. Our choice was to exploit the knowledge about the imaging process to eliminate noise exploiting geometrical information rather than working on histograms. Illumination is another critical issue when dealing with color, as different illuminations can sensibly affect histograms or any other indicator based on colors [29].

Color analysis is seen from a different perspective in this work: instead of exploiting techniques that are very robust to e.g. illumination changes, the focus here is to obtain a very accurate measure in a controlled environment. Differently from mobile robotics applications, the color measurement in a visual inspection system needs to be much more accurate, as wires of very similar colors should be distinguished, and the material of the insulation should be also detected – distinguishing between reflective and matte materials, that show a rather different color signature on the wire. The system deals with some noise factors that are accurately modeled, as they are part of a well-known manufacturing process.

Overall, it can be said that the color analysis algorithm presented here needs to provide very accurate results, based on images taken in a controlled environment. The industrial context also affects the relationship between segmentation and color measurement: for example, in [30] segmentation is driven by color analysis, while in this work the opposite path is exploited: segmentation is achieved by means of background subtraction, and its result drives the analysis of wire color. This approach provides better results, as segmentation is easy to perform and provides accurate results thanks to the *a priori* knowledge of the scene being observed. Color analysis is made complex by its dependency on a number of factors sensibly higher than shape analysis, namely surface roughness, material, and insulation.

This work builds on the preliminary system presented in [31], which has been expanded and thoroughly tested, and is described in detail in this paper. Even though the system presented here relies on some state-of-the-art computer vision techniques, it faces a number of issues that are peculiar to cable crimping visual inspection, like an accurate color measurement in presence of strong noise factors, and the capability of dealing with bent and overlapped wires. To the best of our knowledge, this is the first time a system addressing this task is presented.

III. System description

The cable inspection system described in this paper is designed to be interfaced to a crimping machine by Inarca SpA, Italy. However, its working principle is independent of the specific crimping machine or on the specific crimping process, so it can be applied to any situation in which the sequence of colored wires has to be checked. One of the project requirements was that the crimping machine should not be modified, therefore the visual inspection system had to be installed in the empty space inside the machine itself, as shown in figure 1. While designing the visual inspection system, a set of constraints were imposed in order to integrate the inspection system into the crimping machine, which caused severe limitations on the system geometry.

Visual inspection needs to be performed just after connectors are crimped, while they are being guided out of the machine by means of a metal track. However, such track occludes the connectors, thus the quality inspection needs to be based on the observation of wires only, without relying on the identification of the connectors.

A. System requirements

The goal of the inspection system is to check whether wires are crimped to the connector in the right color order. The check should be performed on all produced items, and the inspection system should be able to keep up with the takt time: for this reason, the computer vision algorithm should run in a time slot shorter than the cycle time, that is about 2-4 seconds per cable for an experienced operator, depending on the number of wires.

The inspection system should be able to handle a number of different situations: the crimping machine can work on several connector types, that differ in the number and size of the wires that can be crimped. Moreover, wires can present insulations of different materials, but with the same color, that have to be distinguished.

The inspection system must be able to deal with any kind of cables, considering that:

- connectors might have different length, but they cannot be directly observed, since they are occluded by a track inside the machine that guides them sideways;
- the number of wires in the connector is variable and depends on the specific production lot;
- no assumptions can be made on the color sequence to be inspected;
- wires can have different insulations, some of which generate specular reflections; wires having the same color, but different insulation materials should be distinguished;

- wires may have markings on them, that can appear in any position;
- wires are often bent by the dragging system that pushes connectors along the track, and they can occasionally overlap;
- wires are of one color, except for the ground wires that present the well-known yellow-green color code;
- some connector configurations include missing wires at certain positions: this absence should be checked.

The need to deal with all these features led to a very general solution which could be effectively exploited also in other application scenarios.

B. Vision system

The vision system is composed by a camera, a LED bar, and a case enclosing them, as shown in figure 1(a). It is installed inside the crimping machine, as it can be seen in figure 1(b). The case is meant to shield from ambient light, thanks to the shadow projected on the rail: this is the reason for keeping the case very close to the analyzed area. The case also holds all the components of the acquisition system while preserving them from dust and other elements – recall that the system is designed to work in an industrial environment. For this reason, there is just one aperture, that lets the camera observe connectors while they are going along the track.

In industrial visual inspection light control is very common, and lighting is therefore part of the system itself [32], [33]. Choices made at this stage strongly influence the image quality and thus the system performance, and should be made in order to minimize the noise sources. In our case, the main source of image degradation is represented by reflections of the light on the wire insulation, because this reduces the area of the wire on which color can be measured. This effect can be reduced by employing diffused light and properly placing the illumination system with respect to the camera.

In the final setup the lighting system is fixed on the front face of the box, at a height of 120 mm to the working plane of the crimping machine, and points perpendicularly to the ground. The LED bar has an illumination area of 82×16 mm, and emits white light, to let the system provide best performance in measuring wire colors. A light diffuser has been placed in front of the LED light source in order to generate diffused light and to avoid reflections from glossy wires. The camera is fixed on top of the box, at a height of 268 mm, and is placed behind the lighting bar, to avoid occlusions.

The camera acquires color images at a resolution of 1600×1200, with a 1/3” sensor. The scene framed by the camera includes the track along which connectors are moved, which is illuminated by the LED bar, and the inner part of the enclosure containing the acquisition system, which is not illuminated. Since the automatic gain control of the image sensor would find a setpoint providing a balance between these two regions, it was disabled and the gain was manually tuned in order to enhance image quality on the illuminated area, where connectors are seen.

One last component of the acquisition system is a tape that has been placed over the metallic working plane of the crimping machine, that would cause a high amount of reflections, therefore saturating the sensor. Such tape has been chosen in an orange color shade not present in the electric wire color coding, and thus works as a contrasting background. In figure 5(a) a sample of acquired image can be seen.

C. Self-learning system

Due to the large variety of configurations to be inspected, a solution in which the operator would have to fully specify the number of the wires, their color sequence, and the presence and/or absence of empty slots in the connector is going to be error prone and tedious to initialize. Even the approach of loading cable specifications from CAD models is not reliable because in the used CAD model the color description is very vague (e.g. red cable) and does not follow any color specification standard.

A self-learning mechanism has therefore been developed: the human operator is asked to correctly crimp a first sample connector, and shows it to the system: this will be considered as the reference, and the system will check that all produced parts have the same color configuration.

Beside the sample connector, during this phase the system analyzes also the color of the background tape, in order to be able to work on backgrounds of any color and pattern. Even though the background is static, this can slightly change during time because of dust or wear, or the tape itself can be changed: background self-learning is then exploited to make the system immune from these aspects. The learning phase is essentially the observation of a single connector that is taken as a model.

During the learning phase, tolerances in color definition are tuned. Learning starts using two sets of threshold triplets, $\{\mathrm{Th_L}\}$ and $\{\mathrm{Th_H}\}$, to be applied on the R, G and B values of wire color. The former set represents a strong constraint, and is employed to verify whether two colors are the same with high confidence: $\{R_1, G_1, B_1\}$ and $\{R_2, G_2, B_2\}$ are considered to be equal if:

$$\{|R_1 - R_2| < \mathrm{Th}_{\mathrm{L},R}\,\mathbf{and}\,|G_1 - G_2| < \mathrm{Th}_{\mathrm{L},G}\,\mathbf{and}\\ |B_1 - B_2| < \mathrm{Th}_{\mathrm{L},B}\}\,. \quad (1)$$

This way a low tolerance is obtained, i.e. two colors are considered to be equal only if the differences in their R, G and B values are very small. The second set of thresholds leads to a higher tolerance. When the colors of two wires are compared, they are automatically considered the same color if their difference satisfies (1), and they are automatically considered as different colors if such difference exceeds $\{\mathrm{Th_H}\}$, that is:

$$\{|R_1 - R_2| > \mathrm{Th}_{\mathrm{H},R}\,\mathbf{and}\,|G_1 - G_2| > \mathrm{Th}_{\mathrm{H},G}\,\mathbf{and}\\ |B_1 - B_2| > \mathrm{Th}_{\mathrm{H},B}\}\,. \quad (2)$$

When the color comparison does not satisfy (1) nor (2), the operator is asked if the two colors should be considered the same or not. This check is performed only once observing the connector model: in other words, the system prompts the user if some wire couples show colors that are similar, but do not look exactly the same. This feature is important when insulations are not perfectly uniform between different wires of the same lot. At the end of the learning phase, color thresholds are further adjusted. The inspection system assumes that the color set does not change along a cable production lot, i.e. it is impossible that colors other than those observed during the learning phase appear during production. Following this assumption, the final thresholds to be employed during production are set to be very restrictive if colors observed during the learning phase are similar, or more relaxed if different colors are strongly different. The final set of thresholds $\{\mathrm{Th}_R, \mathrm{Th}_G, \mathrm{Th}_B\}$ is chosen to be half of the shortest distance between two different colors in the red, green and blue channels, respectively.

The self-learning stage is the only chance the system has to observe one single connector alone, since during production all connectors are side by side and occluded by the track, therefore it is not possible to detect boundaries between them. Thus, this stage is exploited to measure a number of physical parameters, namely:

- width of a single plastic connector,
- number of wires (and possible holes in the wire sequence), and its location with respect to the connector,
- minimum width of a wire,
- distance between any wire pair.

These are all parameters needed in order to understand which is the color sequence that the quality inspection system will assess in the following production lot.

D. User Interface

Our system provides a graphical user interface (GUI), shown in figure 2, carefully designed to ease the communication with the human operator through a touch screen. At the beginning of a new production, the GUI starts in the learning mode, and prompts the operator in order to check that all wires are correctly recognized and labeled. After that, the GUI switches to the inspection mode, and shows connectors being checked. When a component fails the check, the GUI notifies the user with a clear message and a large red icon, see fig. 2. Moreover, the software stops the crimping machine by issuing a signal on a dedicated data line and superimposes the information on the detected defect to the image of the inspected connector, highlighting which part of the component is different from the model; this can be seen in figure 2 in the box labeled "Inspected".

IV. Visual inspection algorithm

The core algorithm of the visual inspection system is composed of two main steps. In the first one, wires are detected and located, so that it is possible to understand if wires are present, and in which positions. Once wires are correctly recognized, the second step, wire color analysis, is triggered.

As previously said, connectors are crimped one at a time by a human operator. After a connector has been crimped, it is pushed by the machine along a track: this causes a sudden motion of all connectors previously produced. The visual inspection system will therefore observe a static scene that sometimes undergoes sudden changes. This is exploited by the system in order to trigger the visual inspection; a frame differencing algorithm similar to that exploited in [34] is used for detecting changes, and understanding when movements are over, in order to trigger a new analysis. This automatic detection of the time instant at which a new connector is ready to be inspected saves the complexity of coupling our software to the electronics of the crimping machine. This also has the advantage of fully decoupling the inspection system from the hardware of the specific crimping machine.

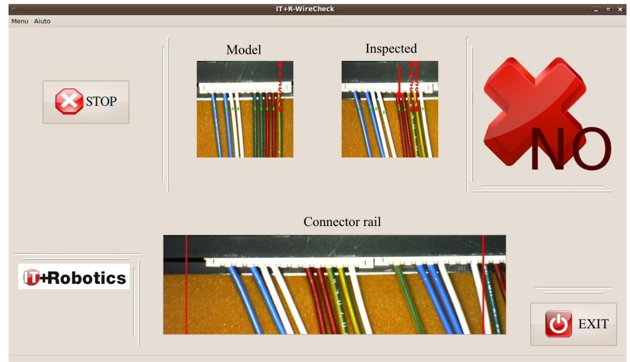


Fig. 2. The appearance of the graphical user interface of the inspection system when a defect is detected.

All the software for visual inspection has been developed in C++ and is based on the OpenCV library[1] [35] for data structures and classes, color conversion and image processing algorithms. The software modules described in the following have been written by the authors from scratch using OpenCV low-level image processing functions. In figure 3 an overview of the whole inspection system is provided; all the modules will be described in detail in the following.

A. Background learning

Before any connector is inserted into the crimping machine, the inspection system acquires an image in which the background is fully visible. Such image will be exploited for performing background subtraction in the wire identification algorithm.

A background equalization process also takes place. Equalization has a number of different meanings in the literature [36]; in our case, it refers to an algorithm that has been developed for correcting uneven illumination in the area framed by the visual inspection machine. In order to reduce noise, a mean shift operator is applied to the background image. The filtered image is then transformed into the HSV (Hue Saturation Value) color space, and the average values for the saturation and value channels are calculated. For each background pixel the S and V channels are compared to the average, in order to understand whether it is lighter or darker. The difference calculated for each pixel is used to obtain an equalized image before the actual inspection process takes place, see figure 5(b).

Information gathered at this stage will be used to correct pixel by pixel each acquired image before the actual inspection

[1] Available at http://www.opencv.org.

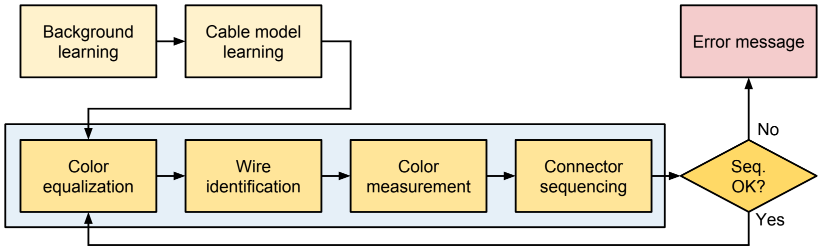


Fig. 3. Overview of the cable inspection algorithm. The loop between color equalization and sequence verification (in the blue box) is triggered when new connectors are pushed along the production line.

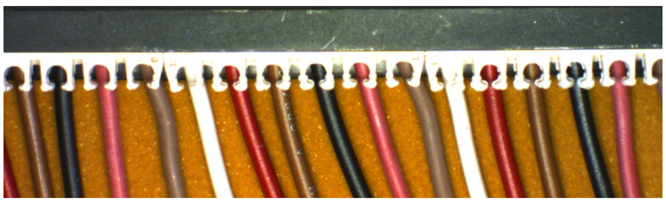


Fig. 4. Input image acquired by the system during the production phase. Note that all cables appear side by side, and the wires are bent.

process takes place.

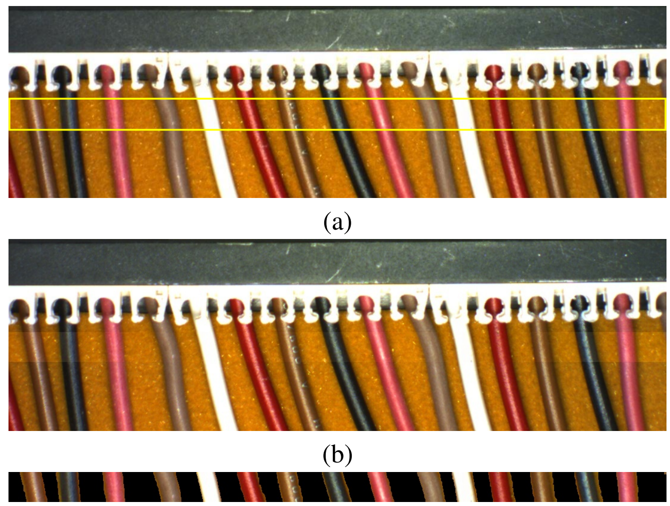
B. Wire identification

Wires are detected as the foreground by a background subtraction algorithm that evaluates the difference between the acquired image, similar to figure 4, with the background acquired before starting the production.

Wire detection is performed considering a ROI (Region Of Interest) in the acquired images, that is placed very close to the connectors (in yellow in figure 5(a)), so that effects due to bending or overlapping are reduced; at the same time, it does not include any part of the connector: this way it is easier to segment wires since they belong to non-connected blobs.

As explained before, an even illumination is desirable; however, mechanical constraints limit the illumination system to be smaller than the observed area, leading to uneven illumination. To compensate for this, information on background illumination, that was measured during the self-learning phase, is exploited to correct foreground pixels. As for the background, this is done by converting the input image to the HSV color space, and correcting the S and V channels by the deviations measured on the background. In figure 5(b) the result of the equalization process is shown: while in the center of the ROI differences with the original image are negligible, the effects are sensibly stronger towards the borders, where illumination is weaker.

In simple cases wires could be located finding connected components in the foreground. This approach is not always working though, since wires can overlap or can appear to be split into two separated regions by highlights. To compensate for these effects, knowledge about wire width acquired in the self-learning phase is exploited, to understand whether connected regions contain single wires or not. The width of each region is evaluated, and compared with the expected value: if they match, a wire has been found. If they do not match, the system runs further investigations.



(a)

(b)

(c)

Fig. 5. Wire identification at various steps: (a) the ROI exploited for wire segmentation, (b) the effect of the equalization process, and (c) the output of the segmentation algorithm.

In case of wire overlap, two possible situations can be observed, as shown in figure 6: wires appear overlapped in the whole region of interest (case A), or they are separated in the upper part of the ROI and then overlap (case B). If it is possible to determine where the overlap starts, the ROI is shrunk in order to include only the area where they can be distinguished. On the other hand, if they appear fused together in the whole ROI, the region is segmented based on the wire width observed in the self-learning phase: the labeled region is divided into smaller regions whose width is equal to the wire width observed in the training phase.

In some cases, strong highlights on the wires are misclassified as background, and the wire is separated into two halves. This results in connected regions whose width is smaller than the one observed in the self-learning phase. When this happens, the system will look for another small connected region at either side, and merge the two together. The wire identification process is summarized in Algorithm 1, where the "cv::" namespace identifier is exploited to indicate that the function is taken from the OpenCV library. Functions without such identifier were specifically developed for this work.

Algorithm 1 Wire identification

```
Output: Set of areas containing the single wires.
  img_diff ← cv::image_difference(img, background)
  contours ← cv::find_contours(img_diff)
  for i = 0 to contour number-1 do
    if contour[i] has dimension of single wire then
      detect single wire in contour[i]
    else
      if contour[i] has Y shape then
        num wires ← Y shape analysis on contour[i]
        sub region ← Y shape segmentation on contour[i]
        for j = 0 to num wires-1 do
          segment single wire on sub_region[j]
        end for
      else
        segment wire group on contour[i]
      end if
    end if
  end for
```

C. Color inspection

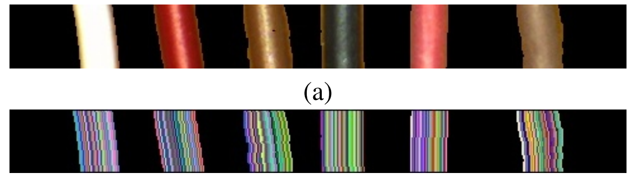
Once wires have been detected, color analysis is triggered after equalizing the image in the HSV domain. Noise factors at this step are highlights and markings on the wires, while shadows do not cause problems thanks to the diffusing shield on the lighting system.

1) Two color wires: Wires are supposed to be of one color, except for ground wires, that are green and yellow. The system determines wire color by applying smoothing filters, as it will be described in the following. However, if this would be applied to a ground wire, its two colors would mix together: to avoid this, the inspection system first checks whether the labelled region contains a ground wire. This check is based on pixel color clustering: since green and yellow have the same (low) blue component and the same (high) green component, and they differ only in the red component, then pixels of the analyzed region are divided into three clusters, G, Y, O (standing for Green, Yellow, Others), using two static thresholds G_{th} and B_{th}, and a dynamic one, R_{th}, defined as $R_{\text{th}} = (R_{\min} + R_{\max})/2$, where $R_{\min}$ and $R_{\max}$ are the minimum and maximum values for the red channel in the considered region. These thresholds were experimentally calculated considering the different production lots for a total of 10000 parts. As a result, a single threshold R_{th} was enough to detect ground wires; however, in different scenarios using two thresholds could also be an option. The generic pixel p_i is then classified following the rule:

$$p_i \in \begin{cases} G & \text{if } R < R_{\text{th}}, G > G_{\text{th}}, B < B_{\text{th}}, \\ Y & \text{if } R > R_{\text{th}}, G > G_{\text{th}}, B < B_{\text{th}}, \\ O & \text{otherwise.} \end{cases} \tag{3}$$

Given the number of pixels for each cluster, N_G, N_Y and N_O, the wire being analyzed is a ground wire if both N_G and N_Y are above a threshold depending on its area. In this case, the analysis ends, since the wire is already classified; otherwise, it is considered a single-color wire, and it undergoes further processing aimed at detecting its color.

Fig. 6. Example of complex situations handled by the system: in case A (on the left) two wires appear overlapped for their whole length, while case B shows two wires only partially overlapped.



(a)

(b)

Fig. 7. The subdivision into longitudinal bins is shown here: segmented wires shown in (a) are divided into bins, as shown in (b): each color indicates a different bin, that goes along the whole wire segment.

2) Color extraction: Wire color measurement needs to be robust to two main noise factors: highlights, that are located at the center of the wire, and markings, that are more difficult to be detected since they can appear anywhere on the wire. These two noise factors are very strong: they can affect a high number of the pixels composing a wire; if the two effects are combined, the number of noisy pixels can be higher than 50% of the total area of the wire. To cope with such high noise levels it is necessary to develop an algorithm that is specific to the problem of wire color detection, i.e. that is based on a simple model of the wire and of the noise factors. In the case presented in this paper, one can observe that both noise factors develop longitudinally. We exploited this observation to create an algorithm for removing such noise factors and obtain an accurate measurement of the wire color even when the color covers less than one half of the observed region. Our solution is that each wire is not analyzed as a whole, but it is divided into longitudinal slices (or bins): this restricts the effect of noise to a small portion of the analyzed region. The number of slices is high: ideally, if a wire has the same width for each row in the ROI, a bin is one pixel width; the number of bins is limited by the minimum width of the wire in the inspection area. In figure 7 an example of wires divided into bins is shown, with each bin represented by a different color.

For each bin, whose spatial domain is D_i, it is possible to understand whether it is affected by markings or not using the index:

$$U = \max_{(x,y)\in D_i} I_g(x,y) - \min_{(x,y)\in D_i} I_g(x,y)\,, \tag{4}$$

where $I_g(x, y)$ is the pixel value in the grayscale image. The U index lets the system understand if pixels in the bin have uniform color or not, and is evaluated in the grayscale image because this eases comparison of contrasting colors which are used for marking the wires. Bins having a U index above a threshold are discarded in further processing, and the same happens to central bins, that are affected by highlights when reflective wire insulations are being inspected. This means the area for color measurement is sensibly smaller than the whole segmented region; however, it is enough to ensure a robust color classification.

To calculate the final color of a wire, defined by the triplet $\{R_w, G_w, B_w\}$, bins are first separately processed by taking the median $\tilde{I}$ of each RGB channel. For the generic bin i the channel R is evaluated as:

$$R_i = \tilde{I}_R(x,y) \text{ for } (x,y) \in D_i\,. \tag{5}$$

Finally, the red value for the whole wire, R_w, is evaluated as the median $\tilde{R}_i$ of the R_i:

$$R_w = \tilde{R}_i \text{ for } i = 0, \ldots, n_{\text{bins}} - 1. \tag{6}$$

The green and blue values, respectively G_w and B_w are calculated in the same way.

It is important to note that the use of the median operator instead of the mean has shown to provide better performance, because it is very robust to color fluctuations at the ends of the histogram: this means that variations in the lightest or darkest parts of the wire, that are often affected by a high noise level, do not impact on the final color measurement. The pseudo-code of the color measurement module is reported in Algorithm 2.

Algorithm 2 Color measurement

Output: The color of a given wire.
  detect clusters
  if size of green cluster $> \mathrm{Th_G}$ **and** size of yellow cluster $> \mathrm{Th_Y}$ **then**
    return ground cable
  else
    n_bins ← evaluate wire bins
    for i = 0 **to** n_bins-1 **do**
      U ← evaluate U index on bin[i]
      if U $< \mathrm{Th_U}$ **then**
        consider Y value in the median evaluation
      end if
    end for
  end if

3) Color comparison: Once wire colors have been extracted, they should be compared with those observed in the model. Instead of using the static threshold sets $\{\mathrm{Th_L}\}$ and $\{\mathrm{Th_H}\}$, the comparison exploits dynamic thresholds chosen during the self-learning phase, and related to the difference between wire colors found in the connector type to be inspected: four threshold are defined, $\mathrm{Th_R}$, $\mathrm{Th_G}$, $\mathrm{Th_B}$, and $\mathrm{Th_{RGB}}$, related to the three RGB channels, and their sum. Two colors A and B, are assumed to be equivalent if their components satisfy the following criterion:

$$\begin{aligned} A \equiv B \text{ iff } (\Delta_R < \mathrm{Th}_R \textbf{ and } \Delta_G < \mathrm{Th}_G \textbf{ and } \Delta_B < \mathrm{Th}_B) \\ \textbf{or } \ (\Delta_R + \Delta_G + \Delta_B < \mathrm{Th}_{RGB})\ , \end{aligned} \tag{7}$$

where Δ_R is the absolute value of the difference between the red channel of the colors A and B: $\Delta_R = |R_A - R_B|$, and Δ_G and Δ_B are analogously defined. The fourth condition, applied to the sum of the three Δ, is exploited for leaving one single channel go out of the boundaries if the two other are very similar: for example, if the threshold is 10 for the single channels and Th_{RGB}=20, the two colors are considered the same even though the difference on one channel is e.g. 15, given that the two other channels together have a difference that is not higher than 5. This was experimentally found to perform better than using higher values for Th_R, Th_G and Th_B.

D. Connectors sequencing

The image processing described so far lets the system inspect single connectors. However, in the production phase connectors are pushed together by the crimping machine. In this situation, there is no way of understanding where each connector starts and ends, because connectors are occluded by the metal rail that guides them to the end of the production line, as described in Section III. The inspection system can rely only on the color sequence in order to assess where a single connector starts.

Wire inspection is triggered when the number of segmented wires in the ROI is at least equal to the number of wires of a single connector. When this happens, the rail is partially empty, and the system compares the pattern detected in the image with the model, starting from the the first wire, that is recognizable.

In a generic image acquired during the production phase it is not possible to make any assumption on the connector position, therefore the leftmost wire in the image is not likely to be the first wire of a connector sequence. To cope with this problem a connector sequencing algorithm was developed. The standard solution of doing a wire tracking would be hard, due to the brisk movements of the connectors along the production line, and it would be computationally very expensive. The connector sequencing algorithm solves the problem of finding a pattern in a given sequence. Every time a connector is produced and pushed along the rail, the whole sequence moves of a length that is not predictable, and the inspection system needs to synchronize again the observed sequence with the model: the sequencing algorithm operates when the scene changes. Sequencing is performed by considering the wires of the model, and matching them in the sequence observed in the analyzed image. Let for example the model M have four wires, labeled as: $M = \{a, b, c, d\}$, while the wires segmented in the image are 10: $I = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. The sequencer will set a reference to the first wire of the image; the first wire of the model, a, is then compared with the reference, 0: if they match, the comparison continues considering the following wires in both model and images, that is: b will be compared to 1; otherwise, the reference is moved, and the comparison is restarted with the set of couples $\{a, 1\}\,\{b, 2\}\ldots$ The algorithm terminates when either the whole model has been successfully compared (that means, a sequence was found), or the reference is moved by a number of positions that is greater than the sequence length, meaning that the observed sequence contains an error. The sequencing algorithm is summarized in Algorithm 3.

Algorithm 3 Connector sequencing

Output: Color sequence, if found.
Input: wires[] {vector of wire colors of the analyzed connector}
Input: model[] {vector of wire colors of the model}
Input: model length {number of colors for each connector}

```
i ← 0
candidate ← 0
while i < model length and candidate + i < model length do
  if wires[candidate+i] == model[i] then
    i ← i+1;
  else
    candidate ← candidate + 1
    i = 0
  end if
end while
if candidate + i == model length then
  return wrong sequence
else
  return sequence found starting at candidate
end if
```

V. Experimental results

The quality inspection system, installed inside a crimping machine, has been tested during real connector production. Tests involved several different connectors types, having a number of wires ranging from 4 to 12, with wire diameter ranging from 1.2 mm to 2.0 mm. Cables used for experiments were taken from 254 production lots, each one composed by 367 parts in average, with a maximum of 1961. Lots exploited for experiments were manufactured by different human operators. Overall, 90266 parts were produced while testing the system, over a period of six months.

A. System performance

To describe system performance, let TP and TN be the true positive and negative, respectively, while FP and FN are the false positive and negative. A true positive (TP) is defined as a sample that is correctly built and that is classified as correctly built by the system; a true negative (TN) is a sample that has a defect, and is correctly classified as a failure by the system. Errors made by the visual inspection system are measured by means of false positives (FP) and false negatives (FN). The former represent the situations in which the system finds a defect in an item that is correctly built; the latter case happens when the inspection system does not recognize a defect and classifies the part as good while a defect is actually present.

Indicators exploited to assess the system performance are accuracy (ACC) and sensitivity or true positive rate (TPR), calculated as:

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \tag{8}$$

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{9}$$

The system has been tuned to adapt to the requirements of this kind of industrial production, for which it is of great importance that no wrongly assembled part is shipped to the customer: to avoid this situation, the system sensitivity should be as high as possible. On the other hand, each false positive causes a warning to be issued to the human operator, who has to manually inspect the part. False positives result in production slowdown, but they do not affect the quality of the produced lot, and are not so critical as false negatives, even if above a certain level can result in the human operator not trusting the system anymore. System tuning affects the thresholds discussed in Section IV that have an impact on how selective the system becomes when comparing wire colors. Modifying such thresholds has an impact on the values of TP, TN, FP, FN: by choosing appropriate value it is possible to maximize the true positive rate by keeping the false negatives low. Since the acquisition conditions (mainly depending on the lighting conditions) are constant, as it is often the case for an industrial system, the parameter tuning does not need to be changed while switching production lot from one to another. All parameters that should be changed are automatically determined in the self-learning stage.

Test results have been divided into two groups, depending on the presence or absence of a ground wire in the observed configuration: this is meant to assess the processing of ground wires, that is based on a technique that is different from the one used for other colors. Experiments were run on 45520 configurations with a ground wire and 44746 without.

For thoroughly testing the visual inspection system a very large set of cables was automatically analyzed and also checked by a human operator at the end of the production line. This is the only way for measuring false negatives, while false positives were easily measured by manually checking all the connectors in which the machine found a defect. To assess the capability of the system of correctly detecting errors, 60 cables were wrongly assembled on purpose in a random way during different production lots; this was made necessary observing that the number of errors that a skilled operator causes was limited to few samples.

Results obtained during the testing phase in terms of false positives and false negatives are summarized in table I, while in table II the values of accuracy and sensitivity are reported. As it can be seen, the system successfully achieves the goal of zero false negatives, leading to a sensitivity of 100%, that is the main feature required for an inspection system working in production. At the same time, the false positive rate is rather low (about 1%), and the level of accuracy reached causes a very small production slowdown. An important result is the correct behavior with ground wires, whose presence has a negligible impact on system performance, as shown in table II.

B. Working examples

Some examples of correct classification in difficult situations can be seen in figure 8: in (a) a production in which brown wires made of different materials are employed is depicted: the system considers the two brown colors as

TABLE I
ERRORS IN TERMS OF FALSE POSITIVES (FP) AND FALSE NEGATIVES (FN) OVER THE TOTAL NUMBER OF ANALYZED CONNECTORS.

	FP	FN	total
Ground	599 (1.316%)	0	45520
No ground	459 (1.026%)	0	44746

TABLE II
SYSTEM PERFORMANCE MEASURED BY ACCURACY (ACC) AND TRUE POSITIVE RATE (TPR) OR SENSITIVITY INDICATORS.

	ACC	TPR
Ground	98.68%	100%
No ground	98.97%	100%

different, and is able to distinguish among them. In (c), a red wire belonging to one of the connectors under analysis presents strong markings, which affect also the area inside the ROI, as it can be seen in the output of the wire identification step (d). The system is nevertheless able to recover the correct wire color, and passes the check. Finally, in (e) a check is made difficult by two factors: on one hand, the ground wire towards the center, beyond being affected by strong highlights, shows a very small green component, looking yellow almost everywhere. On the other hand, the segmentation algorithm, whose output is shown in (f) (binary mask) and (g) (in which the original image and the mask are processed with an AND operator), produces two errors: two wires are not detached in the upper part on the left, while two others are fused together, towards the center; this is because wires are large and put close to each other, producing a high amount of shadow. The system is anyway able to solve such situations, and recover the right wire colors.

The main causes of classification errors are wire markings and highlights not correctly removed. In particular, in certain situations it may happen that a combination of marking and reflections reduces too much the area on which the system can evaluate the wire color, as it is the case of figure 9 (a), (b), (c) and (d).

Another event that might cause false positives is when the background subtraction is not accurate enough, and leaves some background around the wire. In this case wires are separated into blobs slightly larger than they should be. This might lead to a wrong selection of the areas where color is measured, producing classification errors. This case is shown in figure 9 (e) and (f).

Other minor sources of error are:

- the incorrect wire detection, due to problem in the background subtraction phase: this can happen when wires have a color very similar to the background;
- shadows of the wires strongly affecting the background: when this happens, the background subtraction algorithm detects wrong regions due to sudden illumination changes – this is the case of figure 9(e) and (f);
- ground wires placed in a way that lets only one of the two colors to be visible.

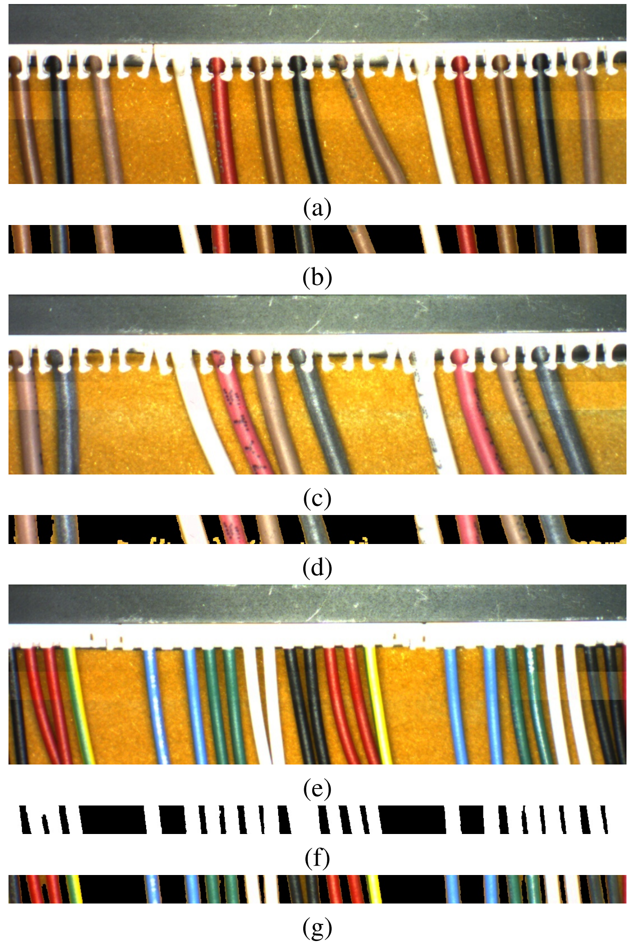


Fig. 8. Example of good behavior in difficult situations: (a) shows an example in which wires of similar colors but different materials (the brown ones) are distinguished by the color analysis algorithm; in (b) the result of the segmentation step can be seen. In (c) the color is correctly measured even though strong markings are present; such markings affect also the ROI in which the analysis is performed (d). In (e) an example of correct behavior with ground wires showing an extremely small green area is shown. Moreover as it can be noticed in (f) and (g), the system can recover two errors of the segmentation step, that, due to shadows, fuses together two wires, either partially (on the left) or completely (towards the center).

C. Computational load

All the computer vision algorithms needed to perform the inspection are run on an industrial computer equipped with an Intel Core2 Duo E7500 processor and 2 GB RAM. On such platform, the average running time is of 1.45 s for each inspection. This processing time is rather stable, and it does not strongly depend on the inspected cable, since the main part is spent in low-level algorithms, that are run before wire segmentation. For this reason, on the cable with the highest number of wires, namely 12, the processing time is of 1.94 s, which is still shorter than the speed at which the human operator produces a cable. No issues have therefore been encountered regarding the computational load.

VI. CONCLUSIONS

In this paper, a system for automatic visual inspection for production of cables with flat connectors has been presented. The system is meant to perform a check on 100% of the

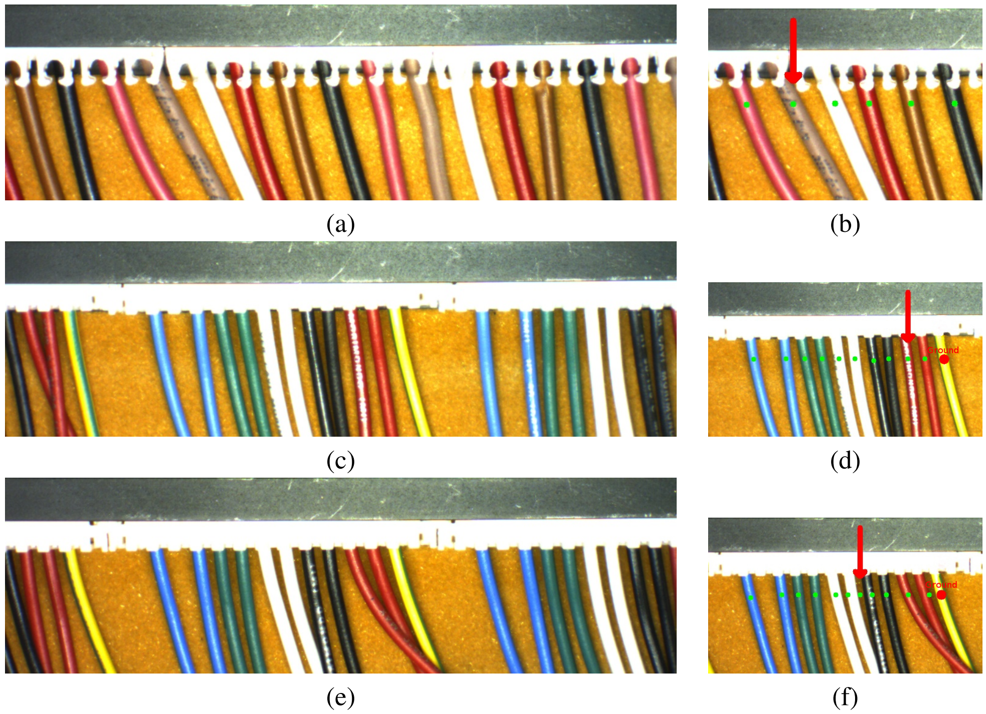


Fig. 9. Examples of wrong classifications: in (a), the system is not able to recover the right wire color, and reports a defect, indicated by the red arrow, as shown in detail in (b); the same happens also in (c) and (d). In (e) and (f), even though the region labeling produces a correct output, and successfully segments the overlapping wires, the presence of shadows around the wires causes their shapes to be enlarged, which in turn leads to wrong localization and color analysis results. In all cases the ground wire is correctly recognized.

produced parts, avoiding any false negative, thus assuring error-free production. The system has a self-learning module that enables it to inspect any kind of cable, with any color sequence, given that a correctly assembled cable can be observed before the production is started.

The system relies on a wire detection algorithm whose output is exploited by a color measurement module, that works on the images after color equalization. The system can deal with difficult situations typical of a real-world scenario: wires that overlap, wires with markings on them, and wires made of different materials. Illumination issues like highlights and shadows are also handled.

The inspection system has been installed into a crimping machine, and tested on a series of real production lots, over a period of several months. The measured performance is very high, since the goal of no false negatives has been reached, with a false positive ratio that is compatible with a production machine. The main sources of error come from two main factors: strong noise on the observed wires, that makes it almost impossible to precisely determine the wire color, and uneven illumination conditions. While the former depends on the raw materials used in the production, the latter effect can be eliminated with a larger observation window, and a stronger illumination. Both could be achieved if the crimping machine would be designed to host the visual inspection system, leaving more room for placing the hardware: the limitations in the current version are due to the fact that the crimping machine was not modifiable and thus the visual inspection hardware had to be installed in a small empty region, the only available in the current version of the crimping machine.

References

[1] T. Newman and A. Jain, "A survey of automated visual inspection," *Computer vision and image understanding*, vol. 61, no. 2, pp. 231–262, 1995.

[2] E. Malamas, E. Petrakis, M. Zervakis, L. Petit, and J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image and vision computing*, vol. 21, no. 2, pp. 171–188, 2003.

[3] M. Shirvaikar, "Trends in automated visual inspection," *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 41–43, 2006.

[4] E. Guerra and J. Villalobos, "A three-dimensional automated visual inspection system for smt assembly," *Computers & industrial engineering*, vol. 40, no. 1, pp. 175–190, 2001.

[5] X. Lu, G. Liao, Z. Zha, Q. Xia, and T. Shi, "A novel approach for flip chip solder joint inspection based on pulsed phase thermography," *NDT & E International*, vol. 44, no. 6, pp. 484–489, 2011.

[6] B. Suresh, R. Fundakowski, T. Levitt, and J. Overland, "A real-time automated visual inspection system for hot steel slabs," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 5, no. 6, pp. 563–572, 1983.

[7] G. Moreda, J. Ortiz-Cañavate, F. García-Ramos, and M. Ruiz-Altisent, "Non-destructive technologies for fruit and vegetable size determination A review," *Journal of Food Engineering*, vol. 92, no. 2, pp. 119–136, May 2009.

[8] M. S. Millán and J. Escofet, "Fabric inspection by near-infrared machine vision," *Optics Letters*, vol. 29, no. 13, pp. 1440–1442, Jul 2004.

[9] S. Ghidoni, M. Minella, L. Nanni, C. Ferrari, M. Moro, E. Pagello, and E. Menegatti, "Automatic crack detection in thermal images for metal parts," in *Heating by Electromagnetic Sources, Proceedings of the International conference on*, May 2013.

[10] T. Ummenhofer and J. Medgenberg, "On the use of infrared thermography for the analysis of fatigue damage processes in welded joints," *International Journal of Fatigue*, vol. 31, no. 1, pp. 130–137, 2009.

[11] C. Neubauer, "Intelligent x-ray inspection for quality control of solder joints," *Components, Packaging, and Manufacturing Technology, Part C, IEEE Transactions on*, vol. 20, no. 2, pp. 111–120, 1997.

[12] B. Jiang, C.-C. Wang, H.-J. Chen, and C.-C. Chu, "Automatic bubble defect inspection for microwave communication substrates using multi-threshold technique based co-occurrence matrix," *International Journal of Production Research*, vol. 48, no. 8, pp. 2361–2371, 2010.

[13] R. Ureña, F. Rodríguez, and M. Berenguel, “A machine vision system for seeds quality evaluation using fuzzy logic,” *Computers and Electronics in Agriculture*, vol. 32, no. 1, pp. 1–20, 2001.
[14] J. Jia, “A Machine Vision Application for Industrial Assembly Inspection,” in *ICMV ’09. Second International Conference on Machine Vision, 2009*. Ieee, Dec. 2009, pp. 172–176.
[15] D. Shumin, L. Zhoufeng, and L. Chunlei, “Adaboost learning for fabric defect detection based on hog and svm,” in *International Conference on Multimedia Technology (ICMT), 2011*, no. 1. IEEE, Jul. 2011, pp. 2903–2906.
[16] P. D’Angelo, C. Wöhler, and K. L., “Model based multi-view active contours for quality inspection,” in *Computer Vision and Graphics*, ser. Computational Imaging and Vision, K. Wojciechowski, B. Smolka, H. Palus, R. Kozera, W. Skarbek, and L. Noakes, Eds. Springer Netherlands, 2006, vol. 32, pp. 565–574.
[17] D. Kumar and K. Kannan, “A Roadmap for Designing an Automated Visual Inspection System,” *International Journal of Computer Applications*, vol. 1, no. 19, pp. 34–37, 2010.
[18] M. Rajeswari and M. Rodd, “Real-time analysis of an IC wire-bonding inspection system,” *Real-Time Imaging*, vol. 5, no. 6, pp. 409–421, 1999.
[19] F. Lahajnar, R. Bernard, F. Pernuš, and S. Kovačič, “Machine vision system for inspecting electric plates,” *Computers in Industry*, vol. 47, no. 1, pp. 113–122, 2002.
[20] T.-H. Sun, C.-C. Tseng, and M.-S. Chen, “Electric contacts inspection using machine vision,” *Image and Vision Computing*, vol. 28, no. 6, pp. 890–901, Jun. 2010.
[21] J. Derganc, B. Likar, and F. Pernuš, “A machine vision system for measuring the eccentricity of bearings,” *Computers in Industry*, vol. 50, no. 1, pp. 103–111, 2003.
[22] D.-B. Perng, C.-C. Chou, and S.-M. Lee, “Design and development of a new machine vision wire bonding inspection system,” *The International Journal of Advanced Manufacturing Technology*, vol. 34, no. 3, pp. 323–334, May 2007.
[23] H.-H. Wu, X.-M. Zhang, and S.-L. Hong, “A visual inspection system for surface mounted components based on color features,” in *International Conference on Information and Automation, 2009. ICIA’09*. IEEE, Jun. 2009, pp. 571–576.
[24] J. Derganc, B. Likar, R. Bernard, D. Tomaževič, and F. Pernuš, “Real-time automated visual inspection of color tablets in pharmaceutical blisters,” *Real-Time Imaging*, vol. 9, no. 2, pp. 113–124, Apr. 2003.
[25] M. Swain and D. Ballard, “Color indexing,” *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991.
[26] Z.-C. Huang, P. Chan, W. Ng, and D. Yeung, “Content-based image retrieval using color moment and gabor texture feature,” in *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on*, vol. 2. IEEE, 2010, pp. 719–724.
[27] C. Wengert, M. Douze, and H. Jégou, “Bag-of-colors for improved image search,” in *Proceedings of the 19th ACM international conference on Multimedia*. ACM, 2011, pp. 1437–1440.
[28] J. Yang, C. Liu, and L. Zhang, “Color space normalization: Enhancing the discriminating power of color spaces for face recognition,” *Pattern Recognition*, vol. 43, no. 4, pp. 1454–1466, 2010.
[29] S. Bianco, G. Ciocca, C. Cusano, and R. Schettini, “Automatic color constancy algorithm selection and combination,” *Pattern recognition*, vol. 43, no. 3, pp. 695–705, 2010.
[30] J. Ning, L. Zhang, D. Zhang, and C. Wu, “Interactive image segmentation by maximal similarity based region merging,” *Pattern Recognition*, vol. 43, no. 2, pp. 445–456, 2010.
[31] S. Ghidoni, M. Finotto, and E. Menegatti, “Self-learning visual inspection system for cable crimping machines,” in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, may 2011, pp. 1–4.
[32] A. Thomas, M. Rodd, J. Holt, and C. Neill, “Real-time industrial visual inspection: A review,” *Real-Time Imaging*, vol. 1, no. 2, pp. 139–158, 1995.
[33] C.-S. Cho, B.-M. Chung, and M.-J. Park, “Development of real-time vision-based fabric inspection system,” *Industrial Electronics, IEEE Transactions on*, vol. 52, no. 4, pp. 1073–1079, 2005.
[34] S. Ghidoni, A. Pretto, and E. Menegatti, “Cooperative tracking of moving objects and face detection with a dual camera sensor,” in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 2568–2573.
[35] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.
[36] I. Bockstein, “Color equalization method and its application to color image processing,” *Journal of the Optical Society of America A*, vol. 3, no. 5, pp. 735–737, 1986.

Stefano Ghidoni received the Master degree in Telecommunication Engineering and the Ph.D. degree in Information Technology from the University of Parma, Italy, in 2004 and 2008, respectively.

From 2004 to 2007 he worked in the field of artificial vision applied to the automotive field as a Ph.D. student of the Artificial Vision and Intelligent Systems Laboratory (VisLab), University of Parma, Italy. From 2008 to 2009, he worked in the artificial intelligence field at Henesis s.r.l., Parma, Italy, which is a spin-off company of the Scuola Superiore S. Anna di Pisa, Italy.

In March 2009 he joined the Intelligent Autonomous Systems Laboratory, University of Padova, Italy, where he is currently working as a Temporary Researcher. His current research interests include intelligent video-surveillance systems, people re-identification in 2D and 3D data, camera networks, and visual inspection for industrial production.

Matteo Finotto received his Bachelor degree from the University of Padova in 2007 and his Master degree from the same university in 2009. After that, he started cooperating with IT+Robotics, Padova, Italy, and IAS-Lab (Intelligent and Autonomous System Laboratory) of the University of Padova, developing several projects related to computer vision applied to quality control. In 2011 he started working at IT+Robotics. He is currently involved in the development of an industrial robot simulator and of visual inspection systems.

Emanuele Menegatti is Associate Professor of Computer Science in the School of Engineering of University of Padova. In 2012, Menegatti got the Italian National Scientific Qualification as Full Professor. His research interests are in the field of Robot Vision Perception. In particular, he has been working on omnidirectional vision systems for autonomous robots, on distributed vision systems, on humanoid robot perception, on 2D and 3D industrial robot vision and automatic visual inspection, and on RGB-D vision algorithms for mobile robots.

Currently, Menegatti is teaching “3D Data Processing” and “Computer Architecture”. Together with prof. Michele Moro, Dr. Menegatti has organized since two years a course on Educational Robotics (Robotica Educativa 2014 - Corso di apprendimento permanente per insegnanti) at the University of Padova, which already formed more than 50 school teachers. Menegatti is European Coordinator of the European project Thermobot. He was Local Investigator for two European Projects 3DComplete and FibreMap. He has been Principal Investigator of several research grants (mainly on robot vision). Menegatti also served as Project Reviewer for the European Commission in FP7. In 2005, Menegatti was a founding partner of IT+Robotics srl, a Spin-off company of the University of Padova, active in the field of intelligent industrial robotics.

In 2014, Menegatti was a founding partner of EXiMotion srl, a Start-up Company of the IAS-Lab, active in the field of rehabilitation robotics and perception for medical applications. In 2014, Menegatti was General Chair of the 13th Intl. Conf. on Intelligent Autonomous Systems (IAS-13). He has served as Associated Editor for Elsevier RAS journal and ICRA & IROS Conferences. Dr. Menegatti published more than 30 journal papers, more than 120 conference papers, 2 books, 4 book chapters, and two patents.