IET Image Processing

Research Article

Row-level algorithm to improve real-time performance of glass tube defect detection in the production phase

ISSN 1751-9659
Received on 11th December 2019
Revised 20th April 2020
Accepted on 12th May 2020
E-First on 9th September 2020
doi: 10.1049/iet-ipr.2019.1506
www.ietdl.org

Gabriele A. De Vitis[1], *Pierfrancesco Foglia*[1] ✉, *Cosimo A. Prete*[1]
[1]*Dipartimento di Ingegneria dell'informazione, Università di Pisa, Via Diotisalvi 2, Pisa, Italy*
✉ *E-mail: pierfrancesco.foglia@unipi.it*

Abstract: In the case of the glass tube for pharmaceutical applications, high-quality defect detection is made via inspection systems based on image processing. Such processing must be fast enough to guarantee real-time inspection and to meet the increasing rate and quality required by the market. Defect detection is complex due to specific problems of the production process: vibration, rotation and irregularity of the tube. All these aspects prevent the efficient use of known techniques. The authors present an algorithm that decreases the processing time of the defect detection phase. The algorithm is based on a moving average filter working at row level, that allows to minimize the effects of rotation, vibration, and irregularity of the tube. Luminosity variations due to the tube curvature are cut by the filter and a threshold algorithm can be applied. They made the evaluation considering different solutions taken from literature. The algorithm outperforms, in processing time, all these solutions with increased accuracy. Experimental measures show that the algorithm achieves a throughput gain of 2.6 times with respect to Canny. They develop also a methodology to get the best values for the algorithm parameters directly at the factory, during the change of production batches.

1 Introduction

Glass tubes that must be later converted into pharmaceutical containers such as vials, syringes, and carpules have historically been made of borosilicate glass [1, 2]. This kind of glass can present defects such as knot inclusions (blobs) due to imperfections in the raw materials used in the furnace, lines of air, or flexible fragments called lamellae and these defects can cause subsequent problems and pharmaceutical recalls [2–6]. Therefore, it is important for pharmaceutical glass manufacturers to check the quality of tubes, with the aim of delivering tubes with defects not influencing the final quality of the product. Such checks can be performed by an inspection system based on machine vision [7], able to recognise defects that are relevant for the specific use of the tube.

An inspection system is depicted in Fig. 1 [8]. Like similar systems [7, 9–11], it is constituted by the image acquisition subsystem and the host computer. The image acquisition subsystem is constituted by a light-emitting diode (LED)-based illuminator, a line scan camera, and a frame grabber, which groups together sequential lines captured by the camera into a single frame, transferring it to the host computer. The system utilises three cameras with related illuminators and frame grabbers to cover a 360° inspection of the tube. The host computer implements defect detection and classification and it takes discard decisions, communicating them to a discarding component. The cutting machine produces tube slices of fixed length (typically of about 1.5 m). Discard decisions are taken considering process parameters, specified via an operator interface or received by the order management system.

The decision to possibly discard a specific slice of the tube can only be made after all the defects detected on the slice have been identified and analysed. This implies that defect detection and classification is subject to strict time constraints. The image acquisition subsystem sends images to the host computer with a rate correlated to the sampling time of the line scan camera. The processing rate of the images during the detection and classification of defects must not be less to avoid overload conditions with consequent loss of some frames. Therefore, the interval between two frames represents an upper limit to the time required to perform the detection and classification of defects. The

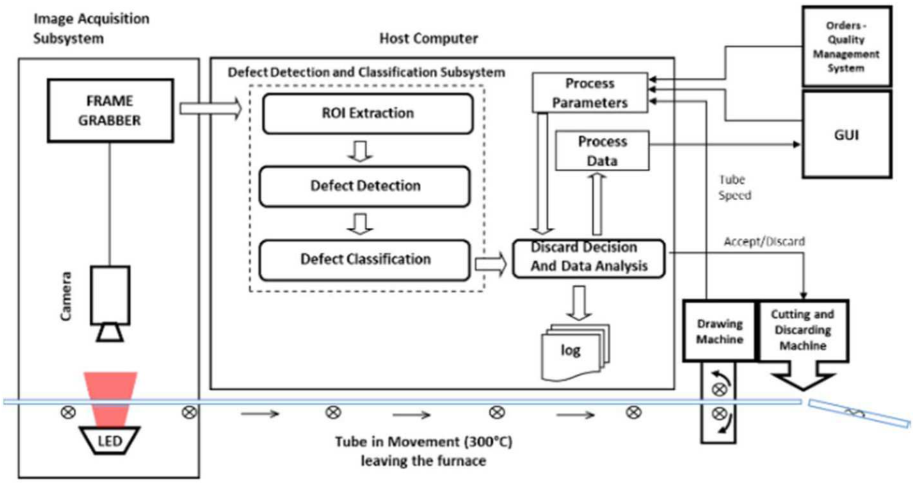


Fig. 1 *Architecture of the inspection system*

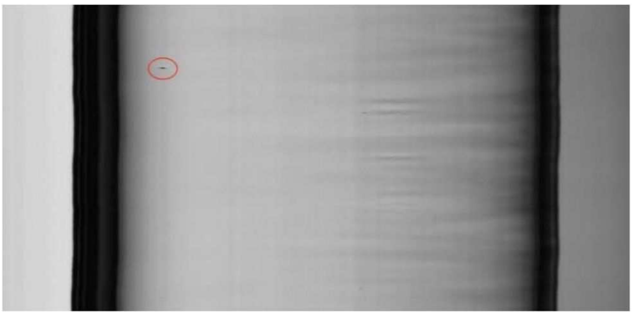


Fig. 2 *Blob defect*

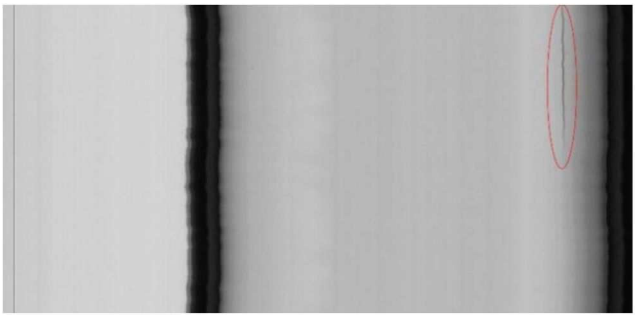


Fig. 3 *Air-line defect. The line, although straight, in the frame appears irregular and curved due to the rotation and vibrations of the tube*

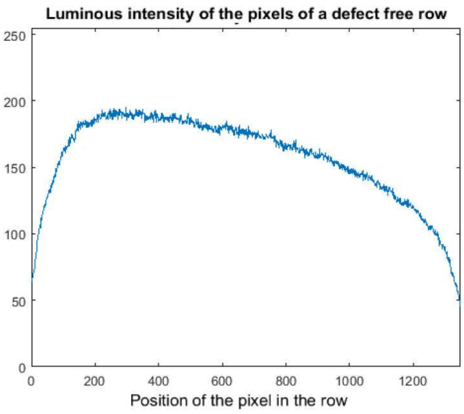


Fig. 4 *Luminous intensity of the pixels belonging to a defect-free row in the frame of Fig. 2*

required reduction of production costs and the required increase in the quality of the defect detection process for a good competitive position on the market lead to the adoption of faster production lines and cameras with greater image definition. This can be achieved by using line scan cameras with reduced sampling times and defect detection and classification algorithms with reduced processing times.

Machine vision has been successfully adopted to perform the inspection in many production processes [7, 12–19]. Anyway, the glass tube inspection task is characterised by specific factors and requirements that do not permit fully exploiting techniques known in the literature. Algorithms based on edge detection [20] can be adopted [8], but are too computationally expensive for the increasingly stringent requirements of such an environment. The glass tube has a cylindrical shape that produces darker areas near the edges of the tube, while the glass is partly suspended and vibrates due to the production process, and in some instants, the illuminator is not perfectly aligned with the tube. Besides, the glass is not perfectly cylindrical but has irregularities that produce a variation of luminosity intensity. As a consequence, algorithms based on global thresholds [21] such as the algorithm proposed by OTSU [22] cannot be adopted due to the cylindrical shape of the tube. Furthermore, since the tubes vibrate and do not have a perfectly circular shape, techniques that exploit background knowledge or other template matching schemas cannot be used [7, 23]. Algorithms based on local thresholds derived from Niblack's [21] can be adopted, but their computational cost can be not adequate to the timing requirement of glass tube production.

The objective of this work is to derive an algorithm to detect defects with reduced processing times compared to the algorithms known in the literature, without reducing the quality of the detection. The achieved algorithm is based on a moving average filter working at the row level, whose combined effects are to cancel the luminous intensity variations due to the curvature of the tube and to limit the effect of rotation and vibration. Via thresholding it is then possible to detect defects. The algorithm is evaluated considering conventional algorithms, such as Canny [20], Niblack [21], or improvements [24]. The results show that the algorithm reduces the processing times by 86% (in the detection phase), the overall throughput increases by 2.6 times, with greater accuracy in detecting defects. We characterise also the dependencies of detected defects and their features by the algorithm parameters and exploit them to develop a tuning algorithm that gets the best values for the algorithm operating parameters, directly at the factory, during the change of production batches.

The paper is organised as follows: in Section 2, we describe the inspection task, in terms of defects that must be detected and the effects of the production process on the acquired images; the quality and performance requirements, and the main settings of the system. In Section 3 we describe the state-of-the-art, and in particular, the typical stages for achieving defect detection, the solutions proposed in the literature, and how such solutions may be applied to our inspection task, and in Section 4 we present the rational of the proposed solution. In Section 5 we describe the proposed algorithm, the strategy adopted for classification, and we present a methodology to tune the algorithm parameters. In Section 6, we first describe the system set-up, the benchmark, and the metrics utilised in the evaluation, then we present and discuss the results.

2 Inspection task description and requirements

2.1 Inspection task description

The main types of defects that can occur in the production of pharmaceutical glass tubes [1, 8], relevant for their critical dimensions and for the significant impact they can have on the final quality of the product are

(i) Knot inclusion (blobs), due to imperfections in the raw materials used in the furnace appears on the captured image as dark patches (Fig. 2).
(ii) Air-lines due to the presence of air bubbles in the furnace, which are pulled by the drawing machine; they appear as darker lines of long dimensions (Fig. 3) with a back illuminator, with the end parts thinner than the centre one. This line, when it is too close to the tube surface, breaks to become thinner and more difficult to detect.

In the pharmaceutical use of tubes, the portions of the tube that present air-lines cannot be utilised for some applications, i.e. in syringes for vaccines, as they can cause contamination of the content. The blob is related to the cosmetic look of the tube, and, in the case of vials, can induce in consumers the idea of the low quality of the product contained.

The inspection task is characterised by the following factors: (i) the cylindrical shape of the glass produces darker areas near the edges of the tube profile; (ii) the glass is partly suspended, and vibrates due to the production technique and, in some instants, the illuminator is not perfectly centred with the tube. By considering only the relevant section of the tube, the luminous intensity of a row presents the increasing variation in the proximity of the edge of the tube, eventually not symmetrical due to the not aligned illuminator (Fig. 4). A row including a defect shows marked downward fluctuations compared to the profile of a defect-free line (Fig. 5). Besides, the glass is not perfectly cylindrical but has irregularities that produce a variation of luminosity intensity (right

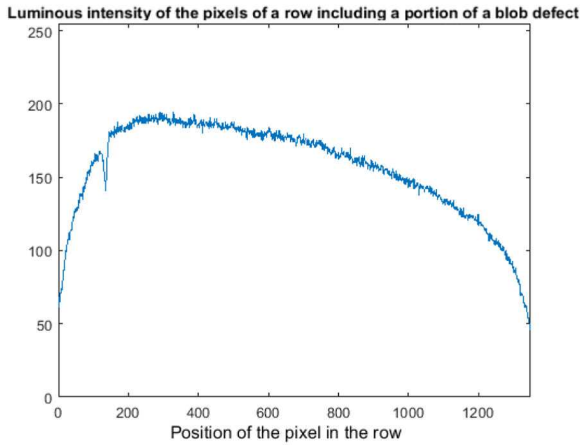


Fig. 5 *Luminous intensity of the pixels belonging to a row including a portion of a blob defect (Fig. 2)*

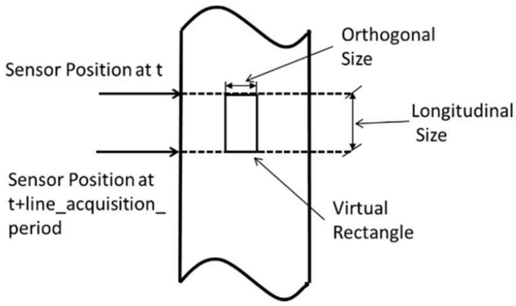


Fig. 6 *Each sensor in the camera catches the light that from the illuminator goes through a virtual rectangle of the tube. Such a rectangle has different orthogonal and longitudinal sizes*

side of Fig. 2). These factors must be considered when choosing or implementing a defect detection algorithm.

2.2 Quality and performance requirements

The line scan camera has a single array of sensors (line array), and a frame 2D image is built by collecting sequential line scans when the tube is moving. Each sensor in the line array covers a virtual rectangle on the glass tube (Fig. 6). The virtual rectangle has a dimension in the orthogonal direction of the tube (orthogonal size) that can be derived from sensor size and optical magnification (1), and a dimension in the longitudinal direction (longitudinal size) that can be calculated from tube speed and line acquisition period (2)

$$\text{orthogonal size} = \text{sensor size} \times \text{magnification} \tag{1}$$

$$\text{longitudinal size} = \text{tube speed} \times \text{line acquisition period} \tag{2}$$

The area of the virtual rectangle covered by one sensor is used to derive the size of the defects observed in the images. The virtual rectangle represents a pixel in the 2D image.

As for the requirements, given the structure of air-lines that due to vibration and rotation change their thickness, achieving a high orthogonal resolution (orthogonal size) is much more important than a longitudinal one (longitudinal size). For glass converting companies, the vision system must identify defects with an orthogonal resolution of 15 µm, and with a longitudinal resolution of 0.5 mm at the maximum tube speed of 4 m/s. The aim is to increase (up to double) the speed of the tube while keeping a resolution.

The cameras are placed near the drawing machine that pulls the tube. Afterwards, the tube undergoes a cutting device that produces a tube slice; the subsequent discarding device must receive the decision to accept/discard taken by the inspection system. Such a decision for each portion of the tube must be taken before it arrives at the discarding device. To ensure this, it is compulsory to process each frame in parallel with the acquisition of the next frame and to satisfy the following constraints:

$$\text{frame processing time} < \text{acquisition period} \tag{3}$$

The acquisition period depends on the number of lines in a frame and the line acquisition period (4)

$$\text{acquisition period} = \text{number of lines in a frame} \times \text{line acquisition period} \tag{4}$$

To satisfy the requirements on orthogonal resolution (15 µm) and longitudinal resolution (0.5 mm), we adopted a 2 k charged-coupled device (CCD) camera, with CCD sensors size of 7 × 7 µm that scan with a frequency of 8 kHz (corresponding to a line acquisition period of 125 µs per scan). The optical magnification has been set to about 1:2.15. From these settings, according to (1) and (2), we can derive that the orthogonal size is about 15 µm due to the sensors' size and optical magnification. The longitudinal size is about 0.5 mm when tube speed is 4 m/s

$$\text{orthogonal size (height)} = 7\,\mu\text{m} \times 2.15 = 15\,\mu\text{m}$$

$$\text{longitudinal size (length)} = 4\,\text{m/s} \times 125 \times 10^{-6}\,\text{s} = 0.5\,\text{mm}$$

With the adopted scan frequency and by considering that in our case, a frame consists of 1000 lines, for formula (4) the acquisition period is 125 ms. Consequently, according to formula (3), the frame processing time must be <125 ms.

As the production speed increases (i.e. tube speed increases) to keep the longitudinal resolution constant, according to (2), we must reduce the line acquisition period (4) proportionally (i.e. increase the sample rate of the camera) and therefore the frame processing time must also be reduced (3).

Other requirements are related to the quality of detection, determined by the discard policies. They concern the capability of detecting all the defects without interpreting noise in luminous intensity as defects, and the accuracy in detecting length/area of the defects.

3 State-of-the-art

A commonly adopted procedure for the inspection of glass tubes, as in other manufacturing environments, is carried out in three stages [9, 10, 16]

- Image preprocessing
- Defects detection
- Defects classification

The preprocessing phase prepares the image for the subsequent phases to reduce the errors introduced as a result of the acquisition and identifying the region to be investigated. The algorithms generally adopted are algorithms for noise reduction, contrast improvement [25], and identification of the region of interest (ROI) [26].

ROI extraction is needed as the acquired frame includes also portions of the support on which the object is placed; such support can be covered with dust and fragments that can easily be classified as defects [27]. For the glass tube, in this phase, is identified as the portion of the image related to the internal part of the tube (Fig. 7). The image portions of the tube edges are excluded from the ROI. Such edges, in fact, look dark in the image due to the reflection of the light rays generated by the illuminator, thus not affecting the sensor of the camera. A ROI extraction algorithm has been proposed in [8]. The algorithm, starting from the centre, search on the left and right part of the tube the minimum values of the luminous intensity and their column numbers, then move from them to detect the edge of the not visible portions of the tube.

In the stage of defect detection, techniques are applied to identify portions of the image whose pixels may represent a defect [9, 10, 16]. These portions are extracted via segmentation methods [16], typically utilising edge detection [20, 28], or thresholds [7].

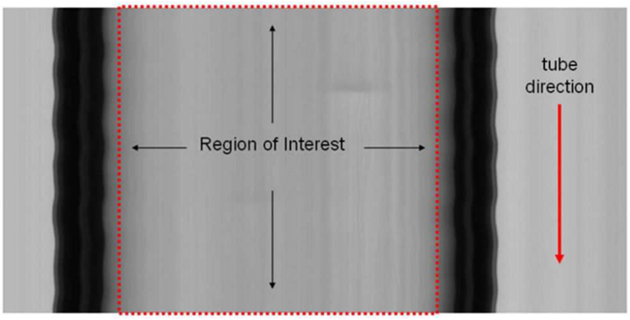


Fig. 7 *Image acquired. In particular, the ROI is highlighted*

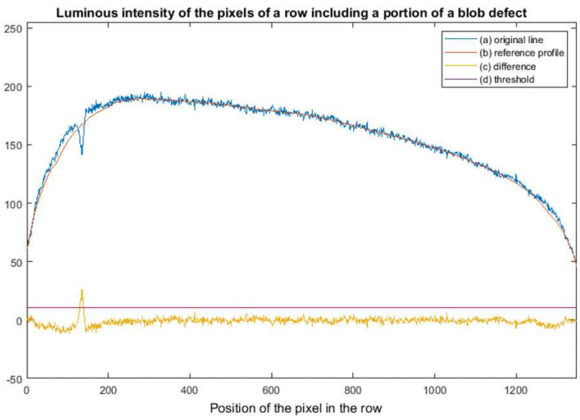


Fig. 8 *Proposed methodology*
(a) For each row of the frame, ***(b)*** A reference profile is derived via a moving average filter, ***(c)*** By subtracting the reference profile to the row, ***(d)*** Defects can be detected applying a threshold

In classification, specific algorithms get and analyse a series of attributes from the segmented regions, possibly combining them within prearranged classes of defects [16].

Edge detection techniques [28] exploit sharp brightness variations in an image to identify defects. Among these techniques, the Canny algorithm [20] is considered one of the best algorithms for detecting defects in noisy images [28]. Using this algorithm, various inspection systems have been implemented, both to detect defects during the production of the glass tube [8] and for detection in other types of glass production [13] and other production sectors [12, 14, 15]. However, the computational complexity of the algorithm may make it unsuitable in future production lines.

Threshold-based techniques can use (i) a global threshold, i.e. a single threshold for the entire image, and (ii) a local threshold, i.e. distinct thresholds in the various areas of the image [21].

The OTSU algorithm and its improvements [22, 29] are widely used techniques based on the global threshold. OTSU works well for images whose luminous intensity histogram has a bimodal or multimodal distribution [16]. Based on OTSU, many inspection techniques have been realised: in [17, 18] to detach defects in glass, in [7] it has been applied to float glass, in [16] for the analysis of transparent part. As a consequence of the characteristics of the glass tube, the detection of the defects cannot be obtained through a single threshold or multiple thresholds, since the luminous intensity of the columns that do not include defects is similar or lower than the luminous intensity of the columns that include defects (Fig. 5). Furthermore, since the tubes vibrate and do not have a perfectly circular shape, techniques that exploit background knowledge or other template matching schemas cannot be used [7, 23].

With regard to the techniques based on local thresholds, the Niblack's technique [21] consists in the use of an adaptive threshold, calculated through the standard deviation and the mean values of the luminous intensity of the pixels belonging to a window of assigned size, positioned centred in the pixel. The method is effective in separating the text from the background; it has also been applied in the processing of text-free images [30] and in the industrial field [19]. Several improvements have been proposed; they try to improve detection or reduce processing time [24, 31]. However, also in these cases, the computational complexity may make such a class of algorithms unsuitable in future production lines.

4 Rational of the proposed solution

To achieve an efficient defect detection algorithm, we start from the observation that the luminous intensity of a row becomes darker in the proximity of the edges of the tube and shows a slow variation in the central area (Fig. 4). In frames where defects are present, the luminous intensity shows marked downward fluctuations compared to defect-free rows (Fig. 5).

Our proposal consists in (i) building the luminous intensity that a row would have in the absence of defects (reference profile) and (ii) subtracting to it the actual luminous intensity of the row, obtaining a signal that significantly varies over zero only in correspondence of a defect (Fig. 8).

A single row cannot be adopted as a reference profile for all the rows in the frame. The tube rotates, vibrates, and in some areas has a not perfectly circular section. These aspects produce significant changes in luminous intensity from one row to another. Anyway, they introduce only low-frequency variations in luminous intensity along a row, while the noise and the defects are a broader band or high-frequency signals added to the reference profile (Fig. 8). The reference profile can then be built by applying a low pass filter to the row. By subtracting the resulting signal to the row, we obtain only the high-frequency components, i.e. noise and defects, and a single threshold can be applied to detect defects. We choose a moving average filter [32] as a low pass filter due to its low computation costs.

The proposed algorithm, moving average glass defects detection algorithm (MAGDDA) calculates, for each pixel of a row ($I(i, j)$ denotes the luminous intensity of the pixel with coordinates i, j), the average value $M(i, j)$ of the window of width WindowSize (WS) centred in the pixel itself (5) and (6)

$$hWS = \left\lfloor \frac{\text{WindowSize}}{2} \right\rfloor \tag{5}$$

$$M(i, j) = \frac{1}{2 \times hWS + 1} \sum_{s = -hWS}^{hWS} I(i + s, j) \tag{6}$$

The difference between $M(i, j)$ and $I(i, j)$ is compared with a threshold k_{thres} and, if greater than it, the pixel (i, j) is marked as belongs to a defect (7)

$$\begin{array}{l} \text{if } (M(i, j) - I(i, j) \ > k_{\text{thres}} \\ \quad \text{then the pixel } (i, j) \text{ belongs to a defect} \end{array} \tag{7}$$

Accordingly, parameters of MAGGDA are the WS and threshold k_{thres}.

5 Algorithm

We assume that the ROI of a frame, extracted using an algorithm known in the literature, consists of grey-scale pixels of R rows and C columns. The algorithm returns an output image O. Pixels of the output image are denoted with $O(i, j)$ and their initial value is white.

The algorithm analyses one row of the image at a time (Fig. 9). The moving average filter of width WS that generates the reference profile is recursively applied in the main loop section. Then, k_{thres} is utilised for defect detection.

For pixels that have a distance from the edge of the ROI less than half WS, a symmetrical window of width twice the distance from the pixel to the edge (start-up loop, final loop) is applied.

```
C = # of columns of Image I
R = # of rows of Image I
hWS = ⌊WindowSize/2⌋
for j = 0, j < R, j++
// Elaboration of the row j

  SUM(0,j) = M(0,j) = I(0,j)

  // Start-up loop
  i = 1
  while 0 < i ≤ hWS
    SUM(i,j) = SUM(i − 1,j) + I(2i,j) + I(2i − 1,j)
    M(i,j) = SUM(i,j)/(2i + 1)
    if (M(i,j) − I(i,j)) > k_thres  then O(i,j) = Black
    i = i + 1
  end while
  // End of Start-up loop

  // Main loop
  while hWS < i ≤ (C − hWS − 1)
    SUM(i,j) = SUM(i − 1,j) + I(i + hWS,j) − I(i − hWS −
        1,j)
    M(i,j) = SUM(i,j)/(2hWS + 1)
    if (M(i,j) − I(i,j)) > k_thres  then O(i,j) = Black
    i = i + 1
  end while
 // End of Main loop

  // Final loop
  while (C − hWS − 1) < i < C
    SUM(i,j) = SUM(i − 1,j) − I(i − (C − i),j) −
        I(i − (C − i) − 1,j)
    M(i,j) = SUM(i,j)/(2(C − i) − 1)
    if (M(i,j) − I(i,j)) > k_thres  then O(i,j) = Black
    i = i + 1
  end while
 // End of Final loop
end for
```

Fig. 9 *MAGDDA algorithm. The algorithm takes as input an image I, made by grey-scale pixels of R rows and C columns, and a threshold k_{thres}. It produces an output image denoted with O of the same size as I*

6 Classification

After applying the defect detection algorithm to classify defects, we build, for each group of adjacent black pixels in the output image, the smallest rectangle that contains them (container of the recognised pixels). The analysis of the containers allows us to detect the real defects, discarding the noise, and classify blobs and air-lines.

To perform classification, we utilise the following parameters:

MA*:* minimum area in terms of number of pixels
$BLHR_{Min}$: blob length and height ratio, minimum value
$BLHR_{Max}$: blob length and height ratio, maximum value
ALLHR: air-line length and height ratio

In particular, MA is the size of the smaller defect that the machine is able to detect. Such size is a quality parameter required by the customer: defects with an area smaller than this value are considered not relevant by pharmaceutical companies. The other parameters are used for classification and derived from the property of the defects to be detected.

For each container, we evaluate the following quantities:

A: area of the container in a number of pixels
LHR*:* length and height ratio of the container

The classification is performed as follows:

1. If $A \geq$ MA, the container is classified as a potential defect.
2. If $BLHR_{Min} \leq LHR \leq BLHR_{Max}$ the potential defect is classified as a blob
3. If LHR $\geq$ ALLHR, the potential defect is classified as an air-line.

The classification algorithm used in the experiments is configured with the following values: MA = 20 pixels, $BLHR_{Min} = 0.1$, $BLHR_{Max} = 0.8$, ALLHR = 2. They were suggested by the experts and derived from the camera settings described in Section 2.2.

6.1 Consideration on WS and k_{thres} and tuning

The parameters of algorithm k_{thres} and WS influence the detection quality. To evaluate such influence, we refer to a frame including an air-line (Fig. 3) with an expected length measured by a human operator (the observed behaviour is common to all the other frames).

The choice of k_{thres} and WS must (i) maximise the number of defects that are properly detected [true positives (TPs)]; (ii) minimise the error in the detected air-lines length; (iii) minimise the number of detected defects that do not represent actual defects [false positives (FPs)]; and (iv) avoid that actual defects are not detected [false negatives (FNs)]. To highlight the dependence between the detected length, the number of FPs and WS and k_{thres}, we have taken a frame that includes an air-line and we have repeatedly applied the MAGDDA and the classification algorithm to it, to determine the detected length. Each time the k_{thres} and WS parameters were changed. We got the following figures: Fig. 10*a* shows in a 3D graph the detected air-line length by varying k_{thres} and WS; Fig. 10*b* shows the error in the percentage of detected air-line length by varying k_{thres} for a set of WS values; Fig. 10*c* shows the detected air-line length by varying WS for a set of k_{thres} values; finally, Fig. 10*d* shows the number of FPs by varying k_{thres} for a set of WS values.

We can observe that

(i) in an interval of WS values the detected air-line length is independent of WS (this interval can be determined in [50, 300]) and such length is determined only by k_{thres} (Figs. 10*a–c*). Outside this interval, the detection performance degrades (Figs. 10*c*).
(ii) The error in the detected air-line length decreases as k_{thres} decreases up to a k_{noerr} value for which the error is null (k_{noerr} is about 5 in Fig. 10*b*). Below this value, the error increases.
(iii) For k_{thres} below k_{nofp} value, FPs are present (k_{nofp} is about 7 in Fig. 10*d*). At this point, there is a small detection error (<10% of the real length in Fig. 10*d*). For k_{thres} over a certain value (about 17 in Fig. 10*b*), no defects are detected (FNs appear).

According to these considerations, the optimal k_{thres} value can be chosen in the interval [k_{noerr}, k_{nofp}] to (i) minimising the number of FPs or the error in detected length and to (ii) avoid FNs.

In an industrial scenario, the MAGDDA parameters depend on some production-related factors (size, diameter, thickness, opacity, and colour of the tube). We have developed an algorithm that permits us to get the best values of the MAGDDA parameters according to the changes in production factors (tuning algorithm). Such an algorithm is executed during the changes of production batches.

The algorithm is based on the premises:

i. a defect causes a high variance value in the columns on which it is placed (variance criteria);
ii. the curvature and the not perfect alignment of the camera and illuminator produce almost linearly change in the column variance;
iii. there is an interval of WS over which the detected defect size is independent of WS value.

The algorithm examines a frame sequence towards the end of a new production batch start-up, and looks for a k_{thres} value that (i)

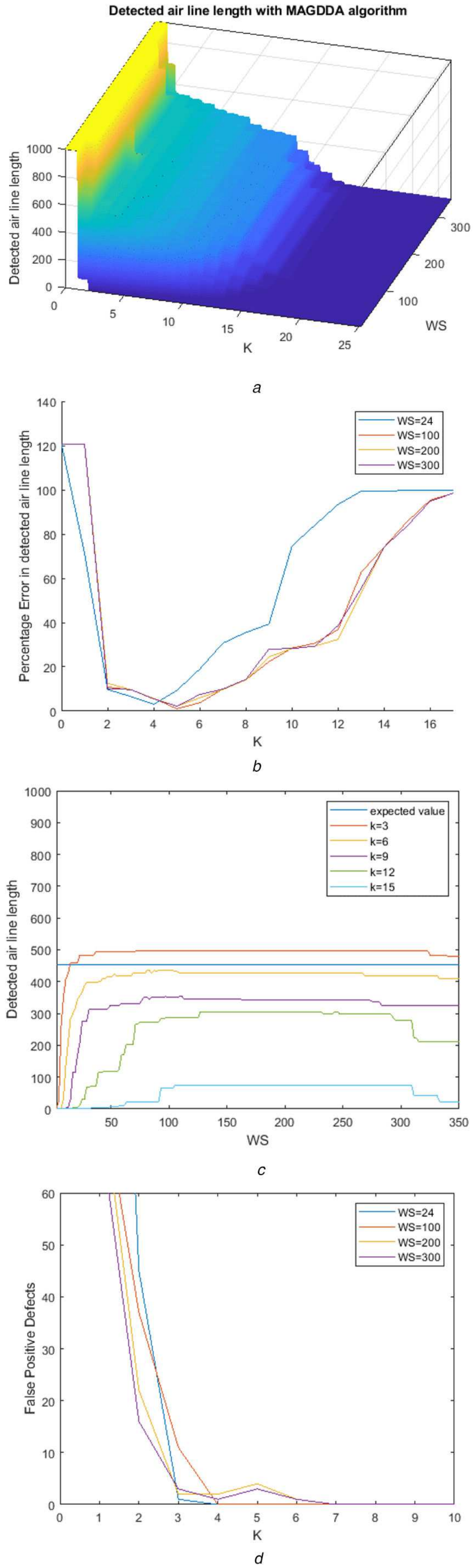


Fig. 10 *Dependence between the detected length, the number of False Positives and the parameters of the algorithm kthres and WS*
(a) 3D graph showing the detected length of an air-line according to k_{thres} and WS parameters, ***(b)*** Error (percentage) in detected air-line length when changing k_{thres} and for WS = 24, 100, 200, 300, ***(c)*** Detected air-line length when changing WS and for k_{thres} = 3, 6, 9, 12, 15, ***(d)*** Number of FPs when changing k_{thres} and with WS = 24, 100, 200, 300

A frame is acquired.
// first section: evaluation of k_{nofp}
Calculates vector V, the variance on the columns of the frame.
Calculates vector T, the linear regression model fitting to V in the least-squares sense.
Calculates vector V* = V-T
k_{thres}=0, ws = 150
Apply the MAGDDA algorithm and classification with < k_{thres}, ws>
while there is at most a defect with V*(j) <2 for each column j of the defect // premise 1
  Increase k_{thres}
  Apply the MAGDDA algorithm and classification with < k_{thres}, ws>
end while
k_{nofp}= k_{thres}
// second section: evaluation of k_{bound}
if there is at least a defect with the last classification
  k_{bound}= k_{nofp}
  SIZE(k_{nofp}) = vector of the sizes of the defects inside the frame detected with threshold k_{nofp}
  SIZE(k_{bound}) = SIZE(k_{nofp})
  while SIZE(k_{bound})$_i$>= SE * SIZE(k_{nofp})$_i$ for each i
    Increase k_{bound}
    SIZE(k_{bound})= vector of the sizes of the defects inside the frame detected with threshold k_{bound}
  end while
  decrease k_{bound}
else
  k_{bound}= Undefined
end if
Return k_{nofp}, k_{bound}

Fig. 11 *Tuning algorithm for a frame*

eliminates FPs and (ii) determines an error in detected defect size lower than a threshold defined by the operator. If such k_{thres} does not exists, it furnishes a list of k_{thres} values and, for each of them, the number of frames for which the condition of zero FPs or bounded length error is not satisfied. The final choice of k_{thres} will be performed by the operator based on the production order requirements.

The algorithm, in the first phase, estimates for each frame, a threshold k_{nofp} that guarantees no FPs. The application of the algorithm to all the frames furnishes a set of k_{nofp} values. The maximum (k_{max}) of k_{nofp} values guarantee no FPs are detected in the frames.

Anyway, in some frames, as k_{max} may be higher than the related k_{nofp}, the adoption of k_{max} can introduce error in detected size. To bind such error for all the frames, in the second phase, the algorithm looks, in each frame, for a new threshold k_{bound}, defined as the largest threshold for which the detected size is less than size factor (SF) times the size detected with threshold k_{nofp}. We call $k_{\text{bound\_min}}$ the minimum of all the k_{bound} values. If $k_{\text{max}} \leq k_{\text{bound\_min}}$, the use of k_{max} guarantees no FP and bounded error. Otherwise, the use of k_{max} may introduce additional error in detection with respect to SF. In this case, the requirement on no FPs may be relaxed, improving the quality of detection at the cost of more FPs.

According to these ideas, the algorithm assumes an initial WS value (i.e. 150 from Fig. 10*c*) and examines a frame sequence. For each frame, it estimates k_{nofp} (Fig. 10*d*). As the algorithm is running in an unsupervised mode, the algorithm evaluates the lowest k_{thres} for which there are no defects detected in low-variance zones (experimentally, a variance value over 2 may indicate the presence of a defect). This suggests—premise1—that such a value of k_{thres} eliminates most or all FPs in the frame, and it is a good approximation of k_{nofp}.

The tuning algorithm is described in Fig. 11. It acquires a frame and calculates vector $\boldsymbol{V}$ of the variance of each individual column, vector $\boldsymbol{T}$ representing the linear regression model fitting to $\boldsymbol{V}$ in the least square sense (premise 2) and the difference $\boldsymbol{V}^* = \boldsymbol{V} - \boldsymbol{T}$. Then applies MAGDDA to the frame with parameters WS = 150, $k_{\text{thres}} = 0$. After the classification, while there is at most a defect with $\boldsymbol{V}^*(j)<2$ for all of the columns j, the algorithm increases k_{thres} and iterates MAGDDA and classification. The loop ends when no defects are detected or all the defects are placed on at least a column j with $\boldsymbol{V}^*(j) \geq 2$. k_{thres} is assigned to k_{nofp}.

In the second section, the algorithm, searches for the threshold k_{bound}, with $k_{\text{bound}} \geq k_{\text{nofp}}$ defined as the largest value for which the size of the detected defect is less than SF times the size detected with threshold k_{nofp}. The algorithm returns k_{nofp} and k_{bound}.

7 Performance evaluation

7.1 System set up, benchmark, and metrics

A prototype inspection system has been implemented, working on a production line of a glass tube foundry. The image acquisition subsystem is composed of a 2K Basler Racer RaL2048 linear camera [33], and a Matrox Solios eCL/XCL-B Camera Link frame grabber with PCI-X and PCIe connections [34]. Illumination is obtained with a COBRA Slim LED Line based on red light [35] in backlight configuration [36]. The illuminator guarantees high power and uniform source of illumination so that changes in the environmental conditions determine no effects on the light on the sensors. The algorithm has been implemented in OpenCV [26]. The communication between the host computer and the cutting and discarding machine is realised via a CIFX 50E-DP board [37].

To evaluate the performance, we used a dataset of 30 frames obtained during production. The dataset includes six air-lines and ten BLOBs; three BLOBs are included in one frame, while two airlines are included in another frame. The number of frames without defects is 17.

Experiments are executed to measure the following parameters: processing time, number and type of classified defects, length of air-lines, and blob area. We perform experiments on a machine like the production one. To evaluate the processing time, we perform 1000 executions of the whole dataset of images [38], and we take the average execution time of each phase and the maximum total processing time. For the number and type of defects, we measured the number of TPs, FPs, and FNs. We considered also the number of defective frames, i.e. the number of frames that contain at least one defect, with a classification like the previous one. Defects have been identified by human experts, with a visual inspection of the tubes.

7.2 Considered algorithms

We have compared MAGDDA with the Canny algorithm [20], Niblack's technique [21], and the Sauvola and Pietikainen [24] improvement of Niblack.

The same ROI extraction algorithm has been utilised in all the algorithms [8]. MAGDDA has been configured with $k_{\text{thres}} = 10.1$, WS = 150, by running the tuning algorithm of Section 5.2 with WS = 150 and SF = 0.8. For Canny, we adopted the thresholds (35 and 80); we take a windows size of 40×40 and $K = -1.7$ for Niblack and $K = 0.05$ for Sauvola. For each of these algorithms, we performed a design space exploration to evaluate the parameter values that permit us to optimise the quality of detection.

Table 1 summarises the main features of the host computer, the algorithms utilised, and the configuration parameters.

7.3 Results

Table 2 reports the number and types of defects detected, the number of defective frames, and their classification. We can observe that the accuracy in defect classification of the MAGDDA and Canny algorithms is similar: MAGDDA has only one more FP than Canny, but the same value for defective frames. In Niblack, there is an increase in FP as tube portions are not perfectly circular, and they produce horizontal darker spots that are detected with the square windows of Niblack. Sauvola works not well near the edge

Table 1 Host computer

Hardware configuration				
Processor	Intel® Core™ i7-940 processor (8 MB cache, 2.93 GHz)			
RAM	8 GB			
defect detection system				
algorithm name	Canny	MAGDDA	Niblack	Sauvola
pre-processing	ROI extraction [8]	ROI extraction [8]	ROI extraction [8]	ROI extraction [8]
defect detection	Canny algorithm	local threshold (Section 4)	Niblack algorithm	Sauvola algorithm
parameters	low-high thresholds (35 and 80)	WS = 145	$N = 20 \times 20$	$N = 20 \times 20$
		$K_{thres} = 10.1$	$K = -1.7$	$K = 0.05$
post-processing	classification (Section 5.1)	classification (Section 5.1)	classification (Section 5.1)	classification (Section 5.1)
implementation	OpenCV	OpenCV	OpenCV	OpenCV

Table 2 Number of recognised defects/defective frames and their classification

	Expected value TP	Canny TP/FP, FN	MAGDDA TP/FP, FN	Niblack TP/FP, FN	Sauvola TP/FP, FN
blobs	**10**	10/5 (0)	10/6 (0)	10/11 (0)	10/655 (0)
air lines	**6**	6/0 (0)	6/0 (0)	6/0 (0)	6/2 (0)
defective frames	**13**	13/2 (0)	13/2 (0)	13/4 (0)	13/17 (0)

Table 3 Air-line length [pixels]

Frame	Expected value	Canny algorithm	MAGDDA	Niblack	Sauvola
1.1	**94**	81	72	73	83
1.2	**483**	473	456	455	464
2.1	**823**	709	758	718	702
2.1	**453**	337	402	353	306
2.6	**634**	502	509	500	477
3.9	**538**	450	513	487	468
cumulative sum	**3025**	2552	2710	2586	2500
cumulative percentage	**100**	84.36	89.59	85.49	82.64
avg abs error, %	**0**	15.42	12.09	15.60	16.76

Table 4 Blob area [pixels]

Frame	Expected value	Canny algorithm	MAGDDA	Niblack	Sauvola
1.3	**135**	270	94	60	124
1.5	**81**	198	74	54	104
1.7	**97**	186	68	47	94
1.9	**100**	132	81	56	112
3.2	**32**	48	29	28	34
3.3	**110**	236	125	83	139
3.5	**1172**	1958	1259	217	1258
3.7	**18**	38	14	24	33
3.7	**43**	55	49	56	79
3.7	**22**	38	24	39	49
cumulative sum	**1796**	3159	1817	664	2026
cumulative percentage	**100**	174.53	100.39	36.69	111.93
avg abs error, %	**0**	81.16	16.36	44.38	38.14

of the ROI, because the average value in the window is low and the deviation is high due to the imperfect circular shape.

Tables 3 and 4 report the air-lines length and the blobs' area of defects that are TP, as detected by an operator and by the algorithms. We report also (i) the cumulative sum of the defects and the fraction of defects (in percentage, row cumulative percentage) detected by the system with respect to the operator measures and (ii) the average of the abs error in percentage (row avg abs error).

All the algorithms have similar accuracy in air-lines length detection. They detect the cumulative length of air-lines with an error of about 15%, and, in each detection, they have an error of 15% on average apart from MAGDDA, which has an error of about 12%. For blobs, the cumulative area is detected by the MAGDDA algorithm almost without error, and in each detection, it has an error of 16% on average. The increased error of Canny for the blobs is caused by the use of the morphological dilate operator to improve the detection of connected regions (useful in air-lines [26] detection). Niblack presents the worst accuracy in blobs area detection: windows including blobs have high values of standard deviation and so the thresholds are low. Sauvola instead has the highest thresholds when the blobs occur, so overestimate them.

Fig. 12 shows the output images obtained when applying the algorithms on a frame including two air-lines. The slopes are opposite because the air-lines are in two radially opposite sections of the tube. We can observe that Niblack detects a high number of noisy points, Sauvola detects FPs near the edge of the ROI. Canny and MAGDDA perform similarly.

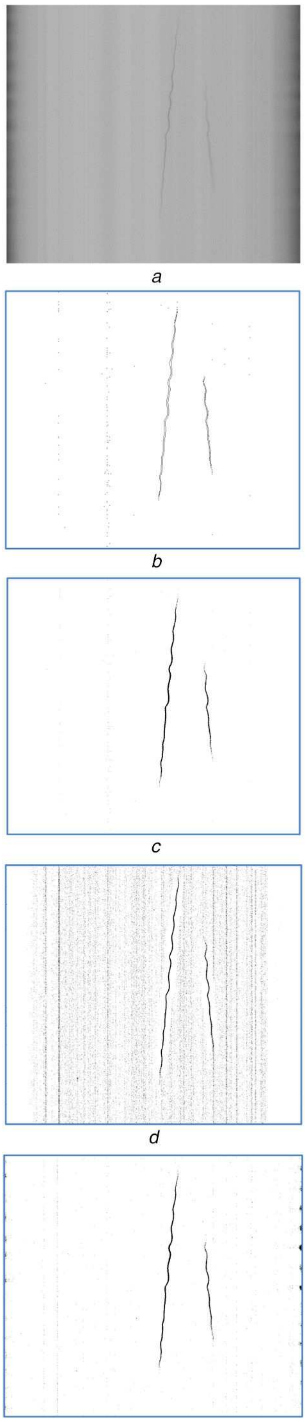
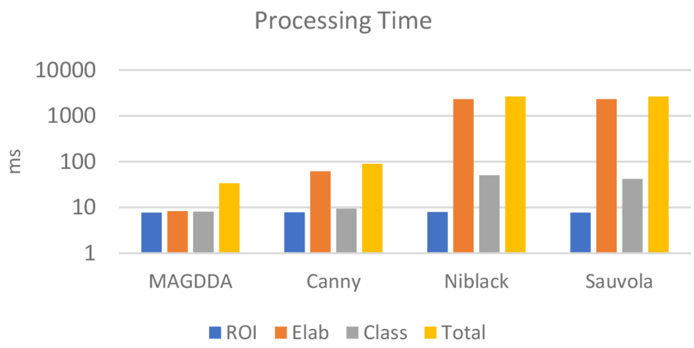


Fig. 12 *The ROI from a frame including two air-lines and output images obtained by the tested algorithms*
(a) ROI Image, ***(b)*** Canny, ***(c)*** MAGDDA, ***(d)*** Niblack, ***(e)*** Sauvola

Fig. 13 reports the measured processing time, highlighting the average time spent in ROI extraction (ROI), defect detection (Elab) and classification (class), and the maximum among the total processing times for each frame in the dataset (total). Elab of MAGDDA is 8 ms while Elab of Canny is 61 ms, with a decrease of 86%. Total for Canny is about 90 ms and for MAGDDA is about 34 ms, with a decrease of 75%. The Niblack's versions considered are not suitable for real-time execution (due to their high computational cost [21]), as they are over 2 s of processing time. Both present a high classification time, due to the high number of noisy point and FP detected (Fig. 12). We tested also an improved version of Niblack that applies an adaptive threshold to each sub-image [31]: processing time is a bit lower than MAGDDA, but the inspection quality is low as, in any case, a fixed threshold cannot be applied also in a sub-image. We report also the throughput in frames per second (FPS) that can be sustained at the maximum execution time. MAGDDA can sustain an FPS of 29.4, which is 2.6 times the fps of Canny.

The proposed MAGDDA algorithm is therefore effective in reducing processing time for defect detection, without reducing the detection quality with respect to algorithms known in the literature. According to (4), the reduction of the processing time allows us to use linear cameras with a higher sampling frequency. Therefore, MAGDDA can be used to improve the quality of inspection, by detecting defects of smaller dimensions, or can be used in faster production lines, maintaining precision and accuracy.

It should be noted that improvement in quality (resolution) and/or production speed, with respect to the parameters used in the study, may result in a change in the lighting system and camera features (resolution and sampling rate), as can be derived from Section 2.1. The proposed tuning algorithm allows the MAGGDA thresholds to be adjusted to changes in defect properties (in terms



	Processing Time				Throughput
Algorithm	ROI	Elab	Class	Total	FPS
MAGDDA	7.693	8.333	8.114	33.941	29.5
Canny	7.845	61.538	9.395	89.824	11.1
Niblack	7.934	2,323.89	50.606	2,642.17	0.4
Sauvola	7.705	2,331.53	41.696	2,645.73	0.4

Fig. 13 *Processing time for an inspection system based on MAGDDA, Niblack, Sauvola, and Canny. The vertical axis is in logarithmic scale. The table below includes also the throughput*

of luminous intensity and number of pixels). The short processing time of MAGGDA allows, as said, increasing the sampling frequency of the cameras keeping the real-time constraints of the inspection system. For example, if the line speed has to be doubled from 4 to 8 m/s, maintaining the resolution requirements of Section 2.2, the camera sampling rate has to be doubled and therefore a frame processing time of 62.5 ms is required (Section 2.2). Of the algorithms considered, only MAGDDA is able to guarantee this level of performance.

8 Conclusion

The quality of the glass tube for pharmaceutical use can be controlled through a vision-based inspection system. The evolution of production processes and the need for greater precision and accuracy in detecting defects increase the frame rate and the number of pixels per frame. Therefore, it is necessary to decrease the time taken to detect the defects even in the presence of an increase in the number of pixels per frame. In the case of glass tubes, the detection is more difficult by specific problems due to vibration and rotation of the tube, and the tube irregularities (imperfect circular shape) which characterise the production process. These aspects prevent the efficient use of techniques known in the literature.

In this study, we have presented and evaluated an algorithm that significantly decreases the processing time of the defect detection phase with respect to algorithms known from literature. The algorithm adopts a moving average filter that works at the row level, minimising the effects of vibration, rotation, and tube shape not perfectly circular. Luminous intensity variations due to the curvature of the tube are cut by the filter, and a threshold algorithm can be applied after filtering to detect defects. The algorithm provides a similar or higher quality of detection than solutions available in the literature (Canny, Niblack, and its improvements) and reduces by 75% the overall processing time. It improves also throughput of 2.6 times with respect to Canny. It can be employed in faster production lines or to increase the image resolution in detection. We have characterised also the dependence between the quality of detection and the parameters of the algorithm. We exploit such dependence to develop an algorithm to get the best values for such parameters directly at the factory, during the change of production batches.

9 Acknowledgments

This work has been partially supported by the Italian Ministry of Education and Research (MIUR) in the framework of the CrossLab project (Departments of Excellence-LAB Advanced Manufacturing and LAB Cloud Computing, Big data & Cybersecurity)

10 References

[1] Berry, H.: 'Pharmaceutical aspects of glass and rubber', *J. Pharm. Pharmacol.*, 2011, **5**, (11), pp. 1008–1023
[2] Schaut, R.A., Porter Weeks, W.. 'Historical review of glasses used for parenteral packaging', *PDA J. Pharm. Sci. Technol.*, 2017, **71**, (4), pp. 279–296
[3] Reynolds, G., Peskiest, D.: 'Glass delamination and breakage, new answers for a growing problem', *Bioprocess. Int.*, 2011, **9**, (11), pp. 52–57
[4] Ennis, R., Pritchard, R., Nakamura, C., *et al.*: 'Glass vials for small volume parenteral: influence of drug and manufacturing processes on glass delamination', *Pharm. Dev. Technol.*, 2001, **6**, (3), pp. 393–405
[5] Food and Drug Administration: 'Recalls, market withdrawals, & safety alerts search'. Available at https://www.fda.gov/Safety/Recalls/, accessed February 2018
[6] Iacocca, R.G., Toltl, N., Allgeier, M., *et al.*: 'Factors affecting the chemical durability of glass used in the pharmaceutical industry', *AAPS PharmSciTech*, 2010, **11**, (3), pp. 1340–1349
[7] Peng, X., Chen, Y., Yu, W., *et al.*: 'An online defects inspection method for float glass fabrication based on machine vision', *Int. J. Adv. Manuf. Technol.*, 2008, **39**, (11–12), pp. 1180–1189
[8] Foglia, P., Prete, C.A., Zanda, M.: 'An inspection system for pharmaceutical glass tubes', *WSEAS Trans. Syst.*, 2015, **14**, (Art. #12), pp. 123–136
[9] Malamas, N., Petrakis, E.G.M., Zervakis, M., *et al.* 'A survey on industrial vision systems, applications and tools', *Image Vis. Comput.*, 2003, **21**, (2), pp. 171–188
[10] Kumar, A. 'Computer-vision-based fabric defect detection: A survey', *IEEE Trans. Ind. Electron.*, 2008, **55**, (1), pp. 348–363
[11] Park, Y., Kweon, I.S.: 'Ambiguous surface defect image classification of AMOLED displays in smartphones', *IEEE Trans. Ind. Inf.*, 2016, **12**, (2), pp. 597–607
[12] Shankar, N.G., Zhong, Z.W.: 'Defect detection on semiconductor wafer surfaces', *Microelectron. Eng.*, 2005, **77**, (3–4), pp. 337–346
[13] Adamo, F., Attivissimo, F., Di Nisio, A., *et al.*: 'A low-cost inspection system for online defects assessment in satin glass', *Measurement*, 2009, **42**, (9), pp. 1304–1311
[14] Huang, X., Zhang, F., Li, H., *et al.*: 'An online technology for measuring icing shape on conductor based on vision and force sensors', *IEEE Trans. Instrum. Meas.*, 2017, **66**, (12), pp. 3180–3189
[15] de Beer, M.M., Keurentjes, J.T.F., Schouten, J.C., *et al.*: 'Bubble formation in co-fed gas–liquid flows in a rotor-stator spinning disc reactor', *Int. J. Multiphase Flow*, 2016, **83**, pp. 142–152
[16] Martínez, S.S., Gómez Ortega, J., Estévez, E.E., *et al.*: 'An industrial vision system for surface quality inspection of transparent parts', *Int. J. Adv. Manuf. Technol.*, 2013, **68**, (5–8), pp. 1123–1136
[17] Huai-guang, L., Chen, Y.-P., Peng, X.-Q., *et al.*: 'A classification method of glass defect based on multiresolution and information fusion', *Int. J. Adv. Manuf. Technol.*, 2011, **56**, (9), pp. 1079–1090
[18] Aminzadeh, M., Kurfess, T.: 'Automatic thresholding for defect detection by background histogram mode extents', *J. Manuf. Syst.*, 2015, **37**, (Part 1), pp. 83–92
[19] Kim, C.M., Kim, S.R., Ahn, J.H.: 'Development of auto-seeding system using image processing technology in the sapphire crystal growth process via the Kyropoulos method', *Appl. Sci.*, 2017, **7**, (4), p. 371
[20] Canny, J.F.: 'A computational approach to edge detection', *IEEE Trans. Pattern Anal. Mach. Intell.*, 1986, **8**, (6), pp. 679–698
[21] Saxena, L.P.: 'Niblack's binarization method and its modifications to real-time applications: a review', *Artif. Intell. Rev.*, 2017, **51**, (4), pp. 1–33
[22] Otsu, N.: 'A threshold selection using an iterative selection method', *IEEE Trans. Syst. Man. Cybern.*, 1979, **9**, pp. 62–66

[23] Kong, H., Yang, J., Chen, Z., *et al.*: 'Accurate and efficient inspection of speckle and scratch defects on surfaces of planar products', *IEEE Trans. Ind. Inf.*, 2017, **13**, (4), pp. 1855–1865

[24] Sauvola, J., Pietikäinen, M.: 'Adaptive document image binarization', *Pattern Recogn.*, 2000, **33**, (2), pp. 225–236

[25] Li, D., Liang, L.-Q., Zhang, W.-J.: 'Defect inspection and extraction of the mobile phone cover glass based on the principal components analysis', *Int. J. Adv. Manuf. Technol.*, 2014, **73**, (9–12), pp. 1605–1614

[26] Bradski, G., Kaehler, A.: '*Learning OpenCV: computer vision with the OpenCV library*' (O'Reilly Media, Inc., USA, 2008)

[27] Matić, T., Aleksi, I., Hocenski, Z., *et al.*: 'Real-time biscuit tile image segmentation method based on edge detection', *ISA Trans.*, 2018, **76**, pp. 246–254

[28] Kumar, M., Saxena, R.: 'Algorithm and technique on various edge detection: a survey', *Signal Image Process.*, 2013, **4**, (3), p. 65

[29] Wakaf, Z., Jalab, H.A.: 'Defect detection based on extreme edge of defective region histogram', *J. King Saud Univ., Comput. Inf. Sci.*, 2018, **30**, (1), pp. 33–40

[30] Farid, S., Ahmed, F.: 'Application of Niblack's method on images'. Int. Conf. on Emerging Technologies, 2009, ICET 2009, Islamabad, Pakistan, 2009

[31] Kulyukin, V., Kutiyanawala, A., Zaman, T.: 'Eyes-free barcode detection on smartphones with Niblack's binarization and support vector machines'. Proc. Int. Conf. on Image Process Computer Vision, and Pattern Recognition, Las Vegas, NV, USA, 2012, vol. 1, pp. 284–290

[32] Smith, S.W: 'The scientist and engineer's guide to digital signal processing'. 1997

[33] Available at https://www.baslerweb.com/en/products/cameras/line-scan-cameras/racer/ral2048-48gm/

[34] Available at https://www.matrox.com/imaging/en/products/frame_grabbers/solios/solios_ecl_xcl_b/

[35] 'COBRATM slim linescan illuminator datasheet'. Available at https://www.prophotonix.com/led-and-laser-products/led-products/led-line-lights/cobra-slim-led-line-light/

[36] Pernkopfa, F, O'Leary, P.: 'Image acquisition techniques for automatic visual inspection of metallic surfaces', *NDT & E Int.*, 2003, **36**, (8), pp. 609–617

[37] 'PC cards cifx user manual'. Available at https://www.hilscher.com/products/product-groups/pc-cards/pci/cifx-50-reeis/

[38] Abella, J., Padilla, M., Del Castillo, J., *et al.*: 'Measurement-based worst-case execution time estimation using the coefficient of variation', *ACM Trans. Des. Autom. Electron. Syst.*, 2017, **22**, p. 4