Nuclear Engineering and Design 335 (2018) 255–264

Contents lists available at ScienceDirect

Nuclear Engineering and Design

journal homepage: www.elsevier.com/locate/nucengdes

Two-phase flow bubble detection method applied to natural circulation system using fuzzy image processing

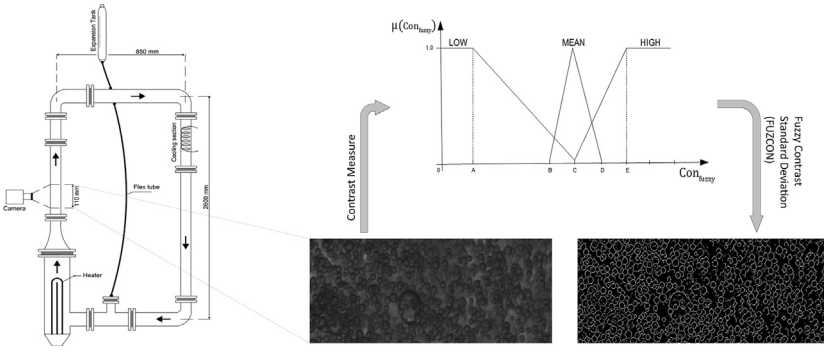
R.C. Bueno[a,*], P.H.F. Masotti[b], J.F. Justo[c], D.A. Andrade[b], M.S. Rocha[b], W.M. Torres[b], R.N. de Mesquita[b,*]

[a] Industrial Process Control, Federal Institute of Education, Science and Technology, IFSP/CPI, Av. Mogi das Cruzes, 1501 Parque Suzano, CEP 08673-010 Suzano, SP, Brazil

[b] Nuclear Engineering Center, Instituto de Pesquisas Energéticas e Nucleares, IPEN/CNEN, Av. Professor Lineu Prestes, 2242 Cidade Universitária, CEP 05508-000 São Paulo, Brazil

[c] Department of Electronic Systems Engineering at the Polytechnic School of the University of São Paulo, CEP 05508-900 São Paulo, SP, Brazil

GRAPHICAL ABSTRACT



ARTICLE INFO

Keywords:
Two-phase flow
Edge detection
Fuzzy system
Image processing
Natural circulation

ABSTRACT

Natural circulation cooling systems are currently used in new nuclear reactors. Over the last decades, research in these systems has focused in the study of flow and heat transfer parameters. A particular area of interest is the estimation of two-phase flow parameters by image processing and pattern recognition using intelligent processing. Several methods have been proposed to identify objects of interest in bubbly two-phase images. Edge detection is an important task to estimate flow parameters, in which the bubbles are segmented to obtain several features, such as void fraction, area, and diameter. However, current methods face difficulties in determining those parameters in high bubble-density two-phase flow images. Here, a new edge detection method is proposed to segment bubbles in natural circulation instability images. The new method (*Fuzzy Contrast Standard Deviation – FUZCON*) uses Fuzzy Logic and image standard deviation estimates of locally measured contrast levels. Images were obtained through an experimental circuit made of glass, which enables imaging flow patterns of natural circulation cycles at ambient pressure. The results indicated important improvements on edge detection efficiency for high void fraction estimation on high-density two-phase flow bubble images, when compared to classical detectors, without the need to use smoothing algorithms or human intervention.

* Corresponding authors.
E-mail addresses: regiscb@ifsp.edu.br (R.C. Bueno), rnavarro@ipen.br (R.N. de Mesquita).

https://doi.org/10.1016/j.nucengdes.2018.05.026
Received 30 May 2017; Received in revised form 9 May 2018; Accepted 25 May 2018

1. Introduction

The cooling system is a critical component of a nuclear power plant, which requires an efficient process of heat exchange. Recently, studies have focused on the phenomenology involved with the nuclear reactor cooling systems and their respective control. A significant portion of new reactor designs uses natural circulation systems for this purpose (Andrade et al., 2000). In such a critical system, it is important to have knowledge about limiting conditions of the two-phase flow regimes to manage heat transfer of the reactor coolant. Within such context, it is of particular interest the study of flow regimes using image processing (Mesquita et al., 2012). In order to determine the void fraction, the image segmentation process can be used to divide the image into subregions to obtain the region of interest (ROI). So, in this paper, edge is regarded as a set of connected pixels about the contour between two regions. Here, a methodology to detect bubble edges in images with high void fraction in a two-phase bubbly flow is presented. It was applied to the natural circulation experimental circuit (NCC) (shown in Fig. 1) in the Nuclear and Energy Research Institute (IPEN), Brazil. This circuit is made of glass, which allows to image flow patterns under ambient pressure.

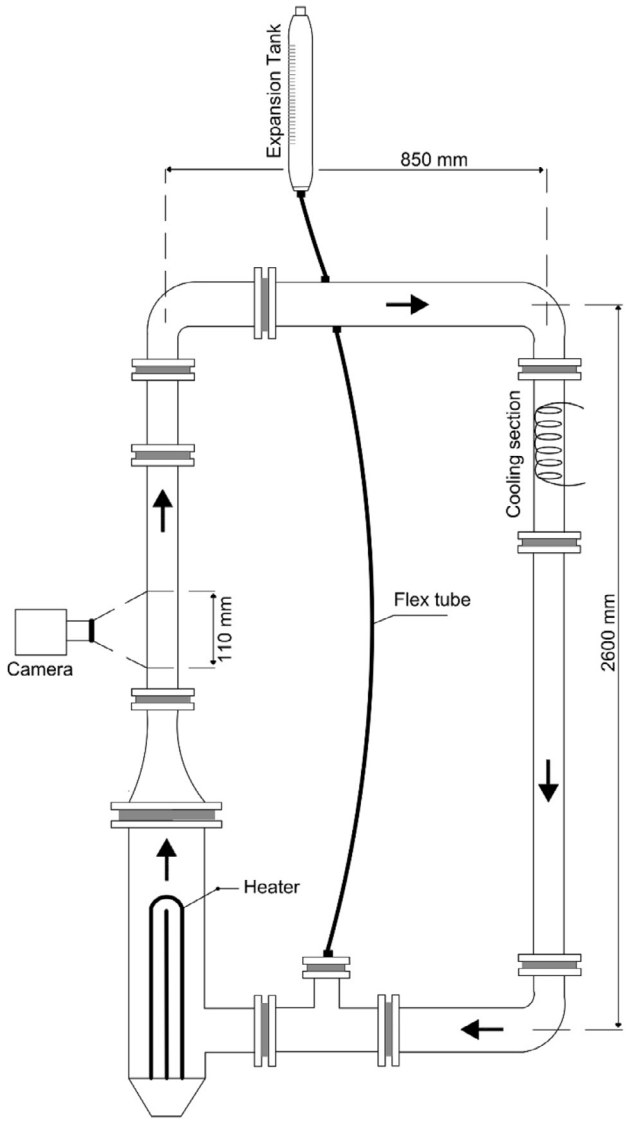


Fig. 1. Experimental setup (NCC) used to capture images on simulation of natural circulation cycles at the Nuclear and Energy Research Institute (IPEN, Brazil).

The paper is organized as follows. Section 1 discusses the conventional method for bubble edge detection, Section 2 details the new methodology, Section 3 presents experimental results using this methodology, and finally Section 4 summarizes the results.

1.1. Bubble edge detection

Two-phase flow pattern transitions have been studied using techniques derived from artificial intelligence or image processing (Mesquita et al., 2012; Abdallah, 2009; Crivalero et al., 2002; Mesquita et al., 2009; Carl and Avdic Senada, 2005; Wu et al., 2006; Sarkar et al., 2005; Zhang and Huang, 1988; Heydarian et al., 2009), such as artificial neural networks, Fuzzy Logic, neuro-wavelet, support vector machine (SVM), and genetic algorithms. Many of those methodologies (Barbosa et al., 2010; Dinh and Choi, 1999; Shi, 2007; Shi et al., 2004; Wenyin et al., 2008) can describe useful patterns by segmenting bubbles and identifying their edges in order to get some parameters, such as bubble area and diameter. However, those methods are limited to bubble edge detection in two-phase flow images, where bubbles are isolated from each other.

There is an increasing demand to improve information on bubble contour to obtain precise parameters in two-phase bubbly flow, such as the void fraction. Void fraction estimation needs the detection of all-round bubble contour and subsequent volume measurement. This detection is challenging when bubbles are close to each other, which can cause over-segmentation (Crivalero et al., 2002; Carl and Avdic Senada, 2005; Wu et al., 2006) due to low contrast present in images, hampering bubble identification (Delacroix et al., 2016; Huang et al., 2007). Other methods (Mishima and Hibiki, 1996; Wang and Dong, 2009) use contrast, entropy, and power with the purpose to identify flow patterns without considering the bubble shape.

In some methodologies that can identify bubbles in high void fraction flow images (Wu et al., 2006; Shi, 2007; Wang and Dong, 2009; Do Amaral et al., 2013; Lau et al., 2013), the edge detection is needed to estimate flow parameters, such as diameter, volume, mass velocity, trajectory geometry, distance between bubbles, and their lifetime. However, those methodologies may fail when intensity changes are abrupt or when objects are very close to each other, as illustrated in Fig. 2. Additionally, lighting inhomogeneity and oscillation are elements that hamper the recognition of objects.

Other methodologies have also been proposed to improve identification of objects of interest in high void fraction two-phase flow images (Wenyin et al., 2008; Barkhoda et al., 2009). They use the Canny operator, combined with other techniques, to establish a smoothing criterion (based on signal noise, good detection, and low spurious response) to identify those objects. However, they can only detect edges in images where bubbles are isolated from each other. Statistical methods, associated with other techniques such as the Watershed method, have been used to deal with the difficulty in detecting heterogeneous light (Zhang and Huang, 1988; Do Amaral et al., 2013; Lau et al., 2013).

Even using statistical methods, some problems, such as image super-segmentation, still arise for heterogeneous lighting. Some investigations have used techniques based on artificial intelligence, such as Support Vector Machine (SVM) (Wu et al., 2006; Sarkar et al., 2005) and Fuzzy Logic (Andrade et al., 2000; Barkhoda et al., 2009; Patel et al., 2011; Jzau-Sheng et al., 1996), using fuzzy inference to assist the segmentation process.

Bhardwaj and Mittal (2012) have reviewed edge detection techniques such as Roberts, Sobel, Prewitt, Canny and especially Declivity ones. Particularly, the Declivity operator has been used in many investigations (Miché and Debrie, 1995; Bensrhair et al., 1996; El Ansari et al., 2010; Cabani et al., 2006). The Declivity proposed in reference (Miché and Debrie, 1995) involves the evaluation of intensity magnitudes of a set of contiguous pixels between two local extremes in an image line. One of the main features of this operator is its ability to

Fig. 2. Grayscale image of a natural circulation two-phase flow obtained at NCC experiment presenting high void fraction.

have a self-adaptive threshold value in an unfiltered image. This threshold value considers the standard deviation of white noise components (Gaussian) multiplied by the change in pixel histogram amplitude in an image profile. The results have shown good performance to find true edges in low-contrast images. Moreover, this method has shown a higher performance compared to classical operators. Declivity method however, is based on a fixed threshold value, that corresponds on value of 5.6 times the standard deviation of the noise component for all images and may induce over-segmentation when is applied using the region growing principle.

This investigation presents a new approach for edge detection, based on a modification of the Declivity operator (Miché and Debrie, 1995), providing precision, speed and global self-adaptive properties to improve void fraction estimation on high-density two-phase flow bubble images. To detect bubble edges, the method is based on a standard deviation estimates over all image (through horizontal and vertical scans) using locally evaluated contrast (gray-level intensity) steps between the local extremes of the Declivity cluster. Subsequently, Fuzzy Logic metrics are considered to elaborate the classifier segmentation rules.

2. Proposed method

The proposed method is based on estimating Contrast steps over intensity values of grayscale images. Initially, a set of k quantized steps is stablished in the [0.0, 1.0] range through locally evaluated standard deviations. Then, two self-adaptive fuzzy membership functions are created based on this set.

2.1. Preprocessing

Images were captured in RGB space (red, green, and blue channels) and transformed into a unique grayscale channel to reduce computational processing. The transformation is done for each image pixel using the conventional relation $L = (299R + 587G + 114B).\ 10^{-3}$, where L is the gray intensity and R, G, and B are the respective color intensities. Each pixel intensity is expressed by an 8-bit depth number which varies from zero to 255, where zero represents 'no intensity' (black) and 255 represents 'full intensity' (white).

In this work, negative images (where gray intensity values are N = 255 − L) were used as reference to associate higher gray intensity levels with border peaks. Global histogram equalization (Gonzales and Woods, 2007) is used in darker images. In this case, it is important to improve gray level distribution, increasing contrast level and consequently providing better edge visualization.

2.2. Contrast methodology definition

The Contrast step measure used in this work is based on an extension of Declivity measure developed in Miché and Debrie works (Bhardwaj and Mittal, 2012; Miché and Debrie, 1995; Bensrhair et al., 1996), emphasizing statistical properties of this feature to object-of-interest segmentation purposes.

Based on an image line, Miché and Debrie defined a basic Declivity (Eq. (1)) as a cluster of contiguous pixels P(x) limited by the $[x_i, x_{i+1}]$ interval:

$$D_i = \{P(x), x \in [x_i, x_{i+1}]\}, \tag{1}$$

where x_i is the position of a i_{th} detected extreme. Miché and Debrie also defined the *declivity amplitude* (d_i) as:

$$d_i = I(x_{i+1}) - I(x_i), \tag{2}$$

where $I(x_i)$ is the grayscale intensity of an extreme pixel.

The extension of Declivity set used at this work was named Mount (M_i). M_i was defined (Eq. (3)) as a cluster of contiguous pixels P(x) limited by an extended interval $[x_{i-1}, x_{i+1}]$, which usually comprises three consecutive extremes:

$$M_i = \{P(x), x \in [x_{i-1}, x_{i+1}]\}, \tag{3}$$

where x_i is the position coordinate of a i_{th} detected peak, and x_{i-1} and x_{i+1} are the positions of the surrounding minima (valleys) as shown in Fig. 3. This interval was chosen to enhance the set of closest pixels to the bubble edges and as consequence, use a smaller number of pixels to compute, implying faster computer processing.

The developed *contrast measure* operator (*Con*) (Eq. (5)) is a non-linear function that estimates the squared grayscale intensity gaps *G(x)* (Eq. (4)) over Mount M_i:

$$G(x) = I^2(x). \tag{4}$$

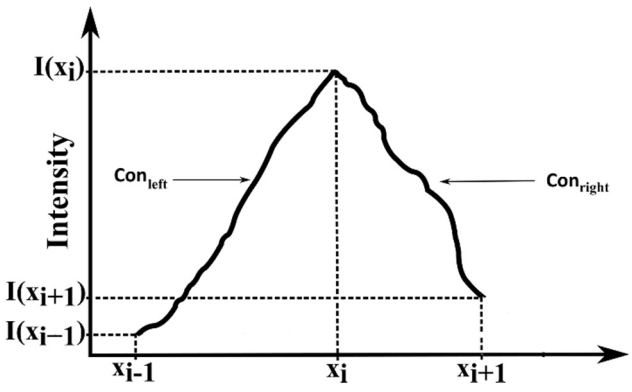


Fig. 3. Left and right contrast measure.

Con is normalized by the difference between the highest and lowest intensities of whole scanned image I(x,y) and defined as:

$$Con = \frac{(\sum_{x\in M_i} |\Delta G(x)|)^{1/2}}{max[I(x,y)] - min[I(x,y)]},\ 0 < Con < 1 \quad (5)$$

The proposed *Con* operator has two new important properties that make pattern detection easier for the current application. The first property is that *Con* may present two opposite values, left (previous) and right (posterior), Con_l and Con_r, shown in Fig. 3. *Con* may also present only one value if the gray levels only decrease or increase throughout M_i.

The second property refers to *contrast measur*e normalization, which by definition, is computed between 0 and 1. This normalization enables this qualitative feature for future use for classification tasks. When *contrast measure* is close to 0, it can be considered very low and possibly irrelevant for edge detection in most cases. On the other hand, when it is close to 1, the *contrast measure* can be considered important for edge detection.

2.3. Contrast measure (Con) implementation basis

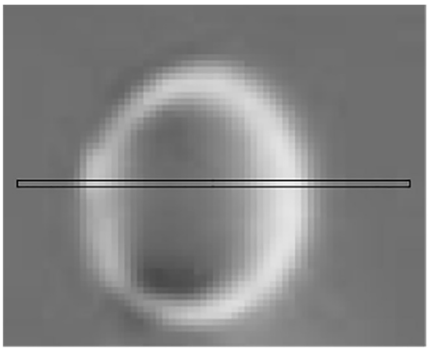
Contrast measure definition considers detected peaks to perform edge feature extraction. *Con* is estimated based on horizontal and vertical rastering. Fig. 4a shows an isolated bubble image example and Fig. 4b shows the highlighted grayscale profile with 1 × 62 resolution.

Fig. 5 shows a set of three different Mounts detected in the highlighted image line shown in Fig. 4. The first Mount (M_1) found is the smallest of all, between pixels 7 and 10. The second one (M_2) is between pixels 10 and 28 and the third one (M_3) is between pixels 29 and 55.

To implement the method, a feature vector (VCD) was created, illustrated in Eq. (6), and consists of four elements. These elements represent respectively the left Contrast value, the peak corresponding pixels column, the peak corresponding pixels and finally, the right Contrast value.

$$VCD = \begin{pmatrix} 0.099,\ 9,\ 0,\ 0.139 \\ 0.581,\ 15,\ 0,\ 0.61 \\ 0.877,\ 44,\ 0,\ 0.855 \end{pmatrix} \quad (6)$$

Mount-1, *Mount*-2, and *Mount*-3 were evaluated by Eq. (3). Both Mount contrast measures Con_l and Con_r can be evaluated with improved computational performance through Eq. (7) to obtain VCD feature vector.



a)

b)

Fig. 4. Bubble image: a) Unit bubble. b) Enlarged bubble linear profile.

Fig. 5. Grayscale [0, 255] intensity values (I(x)) of the detected Mounts (M_i) in range.

$$Con_{l,r} = \frac{\sqrt{I^2(x_{i+1}) - I^2(x_i)}}{\max[I(x,y)] - \min[I(x,y)]} \quad (7)$$

2.4. Contrast standard deviation methodology

Although the lighting used in the capture of the image was constant, there were lighting oscillations due to multiple light incidence angles and, consequently, different refractions caused by the moving bubbles. Contrast inhomogeneity through image occurs due to multiple light refractions and reflections inside two-phase flow. Bubble surfaces and glass tube produce spherical, cylindrical and more complex-shaped lens that cause different capture conditions by the camera apparatus. Thus, the captured images present contrast variations related to lighting inhomogeneity and, therefore, different standard deviation mappings can occur, corresponding to left and right contrast measures. This technique can be used to find standard deviations that represent lighting distribution over bubble contour.

All detected mounts are stored in a N-sized VCD feature vector containing all *contrast measures* (*Cons*) found in the image. A histogram based on *k bins* is constructed quantizing all evaluated *Cons*. This histogram can be considered as a probability distribution as was previously proposed by Otsu segmentation method based on intensity values of image pixels (Otsu, 1979), where contrast probability can be defined as:

$$p_i = \frac{n_i}{N},\ p_i \geqslant 0\ and\ \sum_{i=0}^{k-1} pi = 1. \quad (8)$$

N represents the number of evaluated *contrast measure* values over all image. The n_i variable corresponds to the number of *contrast measures* for each quantized *i* level, where $i \in [0, k-1]$. Based on ten *bins* ($k = 10$) these *Cons* were approximated to discreet values: [0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9] to compose the histogram and improve computational evaluation speed.

Following Otsu procedure, this *Cons* histogram could be divided into two different classes, one associated to the background C_0 and the other to the foreground C_1. The two (w_0 and w_1) classes' occurrence probabilities would be found respectively depending on a *t* threshold level as shown on Eq. (9).

$$w_0 = \Pr(C_0) = w(t) = \sum_{i=0}^{t} p_i \quad and$$
$$w_1 = \Pr(C_1) = 1 - w_0 = \sum_{i=t+1}^{k-1} p_i. \quad (9)$$

The associated conditional probabilities (P_r) for each class enable the μ_1 mean and σ_1 standard deviation estimations for left or right quantized Contrast values as shown on Eqs. (10) and (11):

$$\mu_1 = \sum_{i=t+1}^{k-1} Con_i.\ \Pr(Con_i|C_1) = \sum_{i=t+1}^{k-1} \frac{Con_i.\ p_i}{w_1} \quad (10)$$

$$\sigma_1 = \sqrt{\sum_{i=t+1}^{k-1} (Con_i - \mu_1)^2 . Pr(Con_i|C_1)} = \sqrt{\sum_{i=t+1}^{k-1} \frac{(Con_i - \mu_1)^2 . p_i}{w_1}} \quad (11)$$

In this work, the search for the correct t value was interactively implemented based on a histogram to evaluate standard deviations distribution. A vector $\overrightarrow{w_o}(j)$ was constructed evaluating the cumulative ten ($k = 10$) possible t values of standard deviations associated for background class (C_0), based on a mean value of all detected *contrast measures*. Based on $\overrightarrow{w_o}(j)$, its complement $\overrightarrow{w_1}(j)$ is evaluated. Other two vectors are mounted to further calculations, the vector containing the k quantized contrast measures $\overrightarrow{CON}(j)$ and the number of detected contrast measures for each level $\vec{n}_{CON}(j)$. These vectors are shown on Eq. (12):

$$\overrightarrow{w_0}(j) = [w_0(0), w_0(1), \ldots, w_0(j)]$$

$$\overrightarrow{w_1}(j) = 1 - \overrightarrow{w_0}(j)$$

$$\overrightarrow{CON}(j) = \left[0.0, 0.1, \ldots, \frac{(k-1)}{10}\right]$$

$$\vec{n}_{CON}(j) = [n_{CON}(0), n_{CON}(1), \ldots, n_{CON}(k-1)] \quad (12)$$

where j = 0, 1, …, k − 1, are the possible t values.

The search for the correct (more appropriate) t value can be implemented by an iterative method that utilizes the mean Contrast measure ($\overline{CON}$) evaluated as shown on Eq. (13), to obtain the probability distribution of *Con* values.

$$\overline{CON} = \frac{\sum_{j=0}^{k-1} \vec{n}_{CON}(j) . \overrightarrow{CON}(j)}{N} \quad (13)$$

$$N = \sum_{j=0}^{k-1} \vec{n}_{CON}(j) \quad (14)$$

Based on contrast measures' mean $\overline{CON}$, a vector containing the cumulative means for different t possible values is estimated based on Eq. (15):

$$\overrightarrow{\mu_{1_{Con}}}(j) = \frac{\overline{CON} * \sum_{i=j}^{k-1} \frac{\vec{n}_{CON}(i) . \overrightarrow{CON}(i)}{N}}{\overrightarrow{w_1}(j)} \quad (15)$$

A similar vector for cumulative standard deviations was estimated as described on Eq. (16):

$$\overrightarrow{\delta_{1_{Con}}}(j) = \sqrt{\frac{(\overrightarrow{CON}(j) - \overrightarrow{\mu_{1_{Con}}}(j))^2 * \sum_{j}^{k-1} \frac{\vec{n}_{CON}(i) . \overrightarrow{CON}(i)}{N}}{\overrightarrow{w_1}(j)}} \quad (16)$$

Eqs. (15) and (16) are dependent on $\overrightarrow{w_1}(j)$ values which are the possible foreground cumulative probabilities. It was possible to observe that, in all cases, an abrupt change on $\overrightarrow{w_1}(j)$ occurred for a certain j value. The foreground probability would go to zero when almost no mounts were detected for that contrast level, leading to infinity values for cumulative standard deviation.

A histogram with these cumulative values of $\delta_{1_{Con}}$ was constructed (Fig. 6) to estimate how standard deviation values were distributed over all detected mounts on image. Through these histograms, it was possible to observe that there always happened an abrupt change of the $\delta_{1_{Con}}$ cumulative value corresponding to a j level, where the following values would grow abruptly. That level was called as *abrupt transition* value (*at*) in this work. This was considered as the transition which would characterize the threshold level t for segmentation purposes.

The above histogram was turned into fuzzified values through a recurrence relation designed in this investigation. The standard deviation values correspondent to levels above abrupt transition were approximated to 1. The other fuzzy values were evaluated based on the number of standard-deviation values (S_0) bellow abrupt transition (*at*) that are defined in Eq. (17):

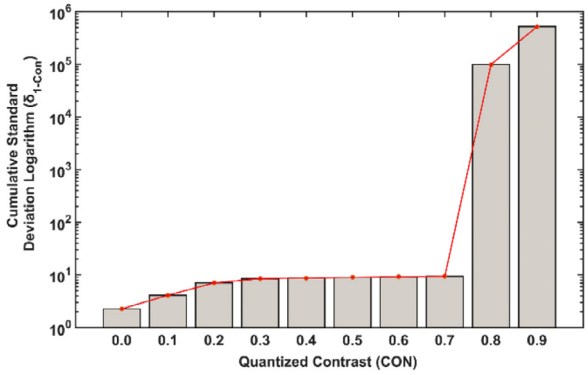


Fig. 6. Standard Deviation Logarithm ($\delta_{1_{Con}}$) histogram of the CON quantized values – the standard deviation *abrupt transition* (*at*) in this example, occurs in 0.7 ($j = 8$, eighth quantized level).

$$S_0 = \text{Std}(0) = \sum_{j=0}^{at} \delta_{1_{Con}}(j)$$

$$Std(j) = Std(j-1) - \delta_{1_{Con}}(at - j + 1) \quad (17)$$

$$Std_{fuzzy}(j) = \frac{Std(j)}{S_0}$$

Eq. (18) presents an example of a typical result in two vectors that represent left and right *fuzzified standard deviations* ($\text{Std}_{fuzzy\text{-}left}$ and $\text{Std}_{fuzzy\text{-}right}$) performed by Eq. (17) over all image. These fuzzified values are obtained and stored for ten discrete (0.0–0.9) values of Con_{fuzzy} to construct a histogram shown in Fig. 7, where all image Mounts are considered.

$$(Con_{fuzzy}, Std_{fuzzy-left}) = \begin{pmatrix} (0.0, 0.03), (0.1, 0.09), \\ (0.2, 0.20), (0.3, 0.32), \\ (0.4, 0.45), (0.5, 0.58), \\ (0.6, 0.72), (0.7, 0.85), \\ (0.8, 1.0), (0.9, 1.0) \end{pmatrix} \quad (18)$$

$$(Con_{fuzzy}, Std_{fuzzy-right}) = \begin{pmatrix} (0.0, 0.01), (0.1, 0.07), \\ (0.2, 0.17), (0.3, 0.29), \\ (0.4, 0.42), (0.5, 0.56), \\ (0.6, 0.70), (0.7, 0.83), \\ (0.8, 1.0), (0.9, 1.0) \end{pmatrix}$$

This approximation was done to be used by Fuzzy Logic System indicating higher standard deviation values in the analyzed image. The transition *at* is used to define the HIGH label to be associated with the higher results for standard deviation values when the membership function is defined.

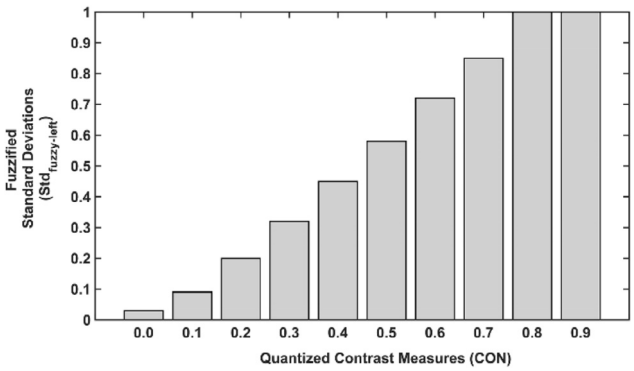


Fig. 7. Fuzzified Standard Deviation ($Std_{fuzzy-left}$) histogram of the CON quantized values.

Table 1
Inferences rules.

Simpler Rules	
Rule 1	IF [CSD(left) is high] AND [CSD(right) is high] THEN (pixel is white)
Rule 2	IF [CSD(left) is high] AND [CSD(right) is mean] THEN (pixel is white)
Rule 3	IF [CSD(left) is mean] AND [CSD(right) is high] THEN (pixel is white)
Rule 4	IF [CSD(left) is mean] AND [CSD(right) is mean] THEN (pixel is white)
Rules with Additional Restrictions	
Rule 5	IF [CSD(left) is mean] AND [CSD(right) is low] AND[μ (CSD(left)) ≥ 0.5] AND [μ(CSD(right)) ≤ 0.5] THEN (pixel is white)
Rule 6	IF [CSD(left) is low] AND [CSD(right) is mean] AND[μ (CSD(left)) ≤ 0.5] AND [μ(CSD(right)) ≥ 0.5] THEN (pixel is white)
Rule 7	IF [CSD(left) is low] AND [CSD(right) is high] AND[μ (CSD(left)) ≤ 0.5] AND [μ(CSD(right)) ≥ 0.5] THEN (pixel is white)
Rule 8	IF [CSD(left) is high] AND [CSD(right) is low]AND[μ (CSD(left)) ≥ 0.5] AND [μ(CSD(right)) ≤ 0.5] THEN (pixel is white)

2.5. Fuzzy contrast standard deviation (FUZCON)

Although each image line has a sequence of *Cons* with different values, it is not possible to detect bubbles based only on this information. Lighting differences during image acquisition, irregular shapes, different bubble sizes, and thicknesses are not fully described by this feature.

Fuzzy Logic is useful to deal with the uncertainty of various thresholds that may be found during bubble edge identification process. The FUZCON method was based on Fuzzy Logic to adapt the contrast values to dynamic range classes which improve computer processing reducing inference rules when compared to methods using fixed range classes. Dynamic range classes were implemented using additional restrictions shown in Table 1 when leading with different contrast ranges.

Following the Contrast properties' feature matrix implementation phase, a fuzzy system was constructed based on fuzzified contrast values Con_{fuzzy} into membership degrees obtained through a fuzzy featured function.

The development of this edge detection system followed several investigations that have explored Fuzzy Logic advantages (Abdallah, 2009; Barkhoda et al, 2009; Patel et al., 2011; Jayachandran et al., 2010). The techniques presented in Barkhoda et al (2009) and Patel et al. (2011) use gradient and standard deviation as parts of two membership functions. Those techniques have shown the effectiveness of using standard deviation to find the gradient of an image. The fuzzy system, developed here, uses a statistical approach of the Contrast Standard Deviation (CSD), labeled Dynamic Fuzzification Base (DFB), which is based on the method described by Barkhoda et al (2009).

The DFB system is a self-adaptive method that analyses each image through dynamic membership functions. These membership functions are necessary to deal with the difficulty in detecting the lighting inhomogeneity and oscillations.

Two membership functions are defined for the left and the right sides. Fig. 8 shows an example of membership functions, described in Eqs. (14)–(16). These functions are defined based on five parameters:

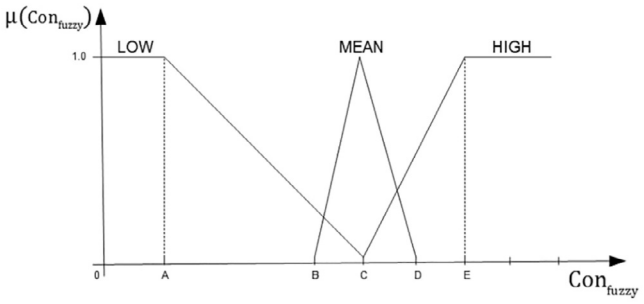


Fig. 8. Dynamic member function.

$(A, B, C, D, E) \in [0.0, 1.0]$.

$$\mu_{Low}(Con_{fuzzy}) = \begin{cases} 0, & Con_{fuzzy} > C \\ \frac{(C - Con_{fuzzy})}{C - A}, & A < Con_{fuzzy} \leqslant C \\ 1, & Con_{fuzzy} \leqslant A \end{cases} \quad (19)$$

$$\mu_{Mean}(Con_{fuzzy}) = \begin{cases} 0, & Con_{fuzzy} < B \\ \frac{(Con_{fuzzy} - B)}{C - B}, & B < Con_{fuzzy} \leqslant C \\ \frac{(D - Con_{fuzzy})}{D - C}, & C < Con_{fuzzy} \leqslant D \\ 0, & Con_{fuzzy} > D \end{cases} \quad (20)$$

$$\mu_{High}(Con_{fuzzy}) = \begin{cases} 0, & Con_{fuzzy} < C \\ \frac{(Con_{fuzzy} - C)}{E - C}, & C \leqslant Con_{fuzzy} < E \\ 1, & Con_{fuzzy} \geqslant E \end{cases} \quad (21)$$

From the standard deviations found for each Contrast (Con_l and Con_r), it is possible to establish the A, B, C, D and E membership function's domain values for each Contrast. They are found by recursive analysis (12), which comprises the self-adaptive method. First, the C parameter is determined to be the closest to 0.5 Contrast Standard Deviation value. In example shown on Eq. (13), these values correspond to 0.4 of (0.4, 0.45) ordered pair in Con_l and 0.5 of (0.5, 0.56) ordered pair in Con_r. Thus, the C parameters values settled to Con_l and Con_r are 0.4 and 0.5 values respectively. Thereafter, the other parameter values are defined as:

- E refers to the highest Contrast Standard Deviation values found in the image, which can be seen in Eq. (13) to be 1.0, corresponding to abrupt transitions;
- B and D are the intermediate values between A and C and between C and E respectively;
- A is the lowest *Contrast Standard Deviation* value found in image, which corresponds to 0.1 as can be seen in Eq. (13), corresponding to 0.09 and 0.07 values.

The fuzzy system's inference rules are shown in Table 1, with four simpler rules (rules 1–4) and four with additional restrictions for CSD_L (CSD(left)) and CSD_R (CSD(right)) values (rules 5–8). Those restrictions were necessary to capture a smaller range to edge detection when the pixel's standard deviation values were classified as Low set in fuzzy rules. Thus, these range allowed reducing the linguistic variable number and consequentially the number of fuzzy rules.

3. Experimental results and discussion

The proposed algorithm Fuzzy Contrast Standard Deviation (FUZCON) was applied to an image database created by instability images of natural circulation flow acquired on an experimental circuit under ambient pressure conditions. Most images were related to different phases called incubation and refill. These phases belong to chugging instability cycles, which are divided in three phases entitled as incubation, expulsion and refill periods, containing unstable flow patterns, described in detail in other investigations (Andrade et al., 2000; Mesquita et al., 2012). This choice of the phases allows observing high densities of bubbles with different sizes. In the expulsion phase, the slug flow is replaced by churn one, when it is not possible to observe bubbles.

The images were obtained using a high resolution professional camera with 21 megapixel CMOS sensor. This camera presents some important features for image processing, such as, highlight tone priority, high ISO noise reduction and lens peripheral illumination correction. An EF 100 mm lens configuration was used: f/2.8L aperture

with Macro IS USM lens, ISO 3200, manual focus, 1/8000 s shutter speed, and remote shutter. Backlight illumination used a 1000 W lamp reflector with dimmer.

The acquisition time interval of one image to another was one second during the experiment. The distance between the camera lens and cylindrical tube were adjusted to obtain an estimated depth-of-field of 2 cm. This controlled depth allowed focused images capturing of the two-phase flow inside the tube. Depth-of-field calibration was done using a focus pattern (ladder) beside the tube. Each ladder step had a 0.5 cm depth spacing. Analyzed images were organized in a database containing 1789 images. These images were cropped and presented in three basic resolutions: 2316 × 1208, 864 × 330, and 744 × 374. Different spatial resolutions were used to evaluate the methodology when applied to different bubble sizes.

The performance of the proposed edge detector was evaluated using receiver-operating characteristics (ROCs) method to compare its performance with some classical border detectors' performance. This comparison was based on images that were manually classified, which are usually called Ground Truth (GT) images. GTs usually indicate an ideal detection observation (Heath et al., 1998). This kind of comparison results in relative rankings that do not change, even with different GTs manually produced by different individuals or at different times. The ROCs points' absolute position may vary but the order of results is stable (Heath et al., 1998).

ROC analysis involves four metrics: true positive (TP), false positive (FP), true negative (TN), and false negative (FN). If a detector reports an edge pixel and GT also reports an edge pixel, TP is increased. If a pixel is reported in a GT area without pixels, FP is incremented. Wherever there are no edge pixels in GT and detector, TN is incremented. If there are pixels in the GT edges but not in the detector, FN is incremented.

In the GT images, the 255 value (white) represents the edge pixels and zero value (black) indicates regions without interest. The classical edge detectors are based on image operators that produce grayscale images within the [0, 255] interval. These 256 values have to be transformed into a binary image classified as edge or non-edge pixel. This is done using a threshold value above which pixels are considered an edge. The classified image, then, is represented only by 255-valued pixels (edge) and zero-valued pixels (non-edge). The chosen threshold value used for the classical edge detectors in this ROC analysis was 50. This choice was done based on an assessment criterion derived from observation that pixels bellow this value were never part of 'objects of interest' (bubbles) within used images.

It is important to emphasize that GTs were produced in this work considering edges that were visually evaluated as 'in focus', as can be observed in Fig. 9 (d and f). Focused bubbles are important to be detected in order to enable further volume void fraction estimates. Images were acquired purposely within a pre-determined depth-of-field as was previously described.

High TP and TN values, associated with low FN and FP values, are required for a detector to be considered as good. When (ROC's graph point is nearer to the (0, 1) point it means a higher TPR/FPR ratio value, and thus an indication of a good classifier. The method is considered random when ROC's values are near to the diagonal line from the left bottom to the top right corner. Fig. 9 shows the grayscale image extracts (with 2316 × 1208, 864 × 330, and 744 × 374 resolutions) of the database captured images and their corresponding GT images.

The detection algorithm was implemented using Python programming language and was applied to several images, in order to be compared with classic edge detectors: Sobel, Laplacian of Gaussian (LOG) and Watershed operators (Sobel, 1974; Marr and Hildreth, 1980; Meyer, 1991). The Sobel and LOG masks are shown respectively in Eqs. (14) and (15), and can be implemented via convolution method in the x- and y-directions.

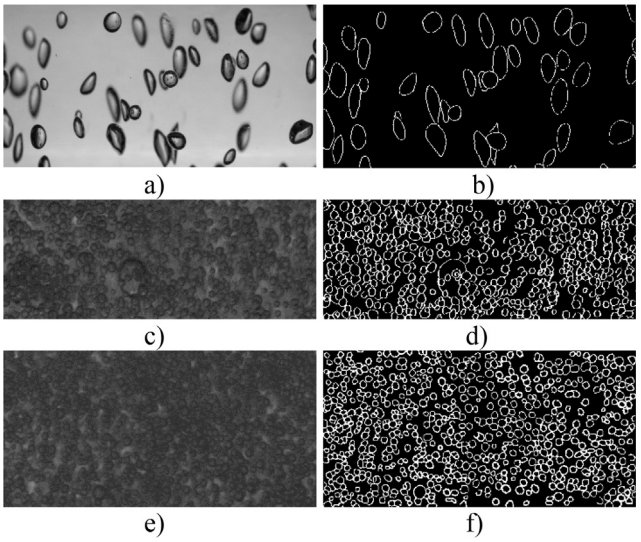


Fig. 9. Ground Truth images (a, c and e) corresponding to Grayscale image extracts (b, d and f) GT with 2316 × 1208, 864 × 330, and 744 × 374 resolutions, respectively.

$$x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{22}$$

$$x = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad y = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \tag{23}$$

The results are shown in Figs. 10–12. Fig. 10 presents detections results of flow patterns with low void fraction two-phase in images. Figs. 11 and 12 show the images of flow patterns with high void fraction from the refill-incubation instability transition phase. These last images (Figs. 11 and 12) present many compacted bubbles, which represent a major challenge for detection algorithms. The proposed algorithm has obtained satisfactory results for all analyzed image types.

The Sobel operator highlights bubbles that can be identified through well-connected pixels presenting different intensities. Stronger intensities are related to main edges but Sobel detector gives several

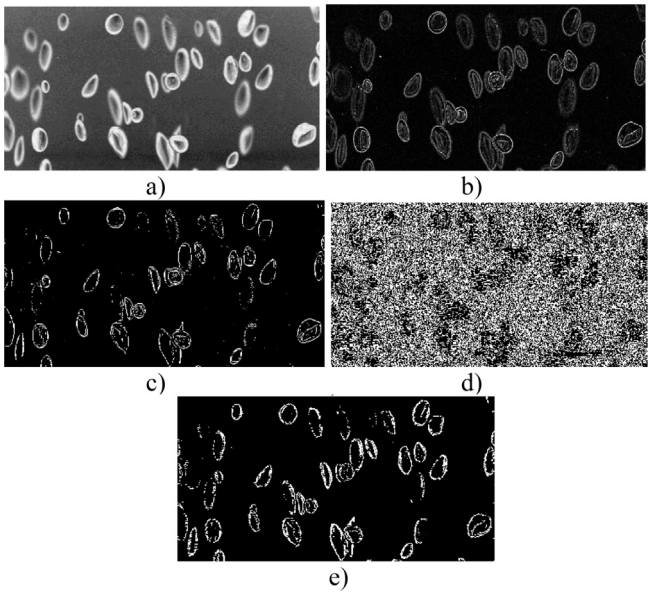


Fig. 10. Simple Image. a) Negative Image. b) Sobel Edge Detector. c) LOG Edge Detector. d) Watershed e) Fuzzy Contrast.

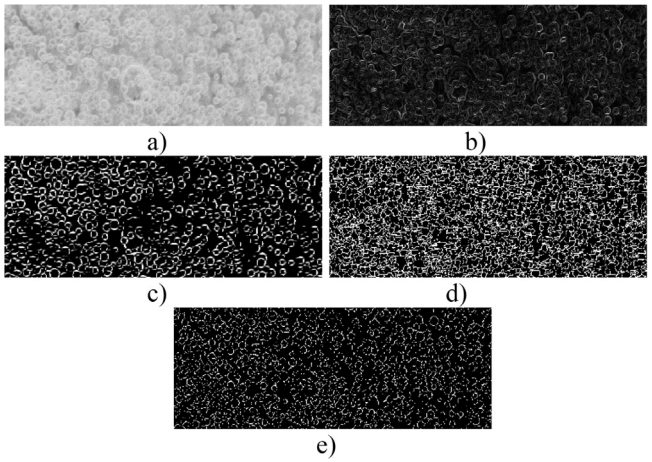


Fig. 11. Refill Incubation Image. a) Negative Image. b) Sobel Edge Detector. c) LOG Edge Detector. d) Watershed e) Fuzzy Contrast.

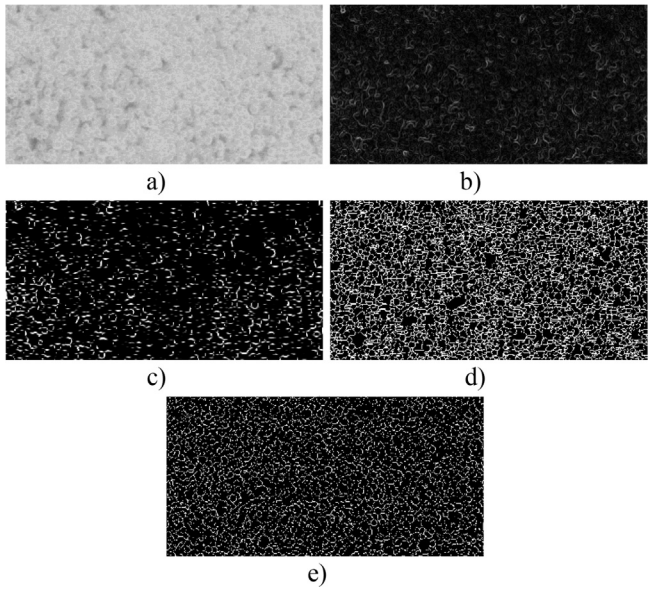


Fig. 12. Refill Incubation Image. a) Negative Image. b) Sobel Edge Detector. c) LoG Edge Detector. d) Watershed e) Fuzzy Contrast.

output details even in image regions with blurred bubbles. This detector also has difficulty in identifying edges where low contrast is present. Additionally, the operator could isolate inner bubble shadows, resembling a double edge, shown in Fig. 10b.

Watershed-based segmentation method consists in finding image features called “water lines” that are defined as the limits of a “flooding process”, when the top-down image topology is filled until a stage is obtained only where dam tops can be observed (Gonzales and Woods, 2007). At this point, theses dam tops represent desired segmentation corresponding the watershed lines.

More stable segmentation results can be obtained through this algorithm when connected segmentation edges are wanted, such as in very close bubble edge detection. However, when this technique (Meyer, 1991) was applied to the images on this investigation, disconnected paths were formed, occurring over-segmentation, as shown in Figs. 9d, 10 and 11. This over-segmentation originated from many tone variations in the background, inside the bubbles and in their edges.

The LOG operator, at its first step, uses Gaussian operator smoothing in order to reduce image noise followed by a Laplacian operator application. This operator is mainly based on the Gaussian filter standard deviation, that is, the higher the value of this standard deviation, the

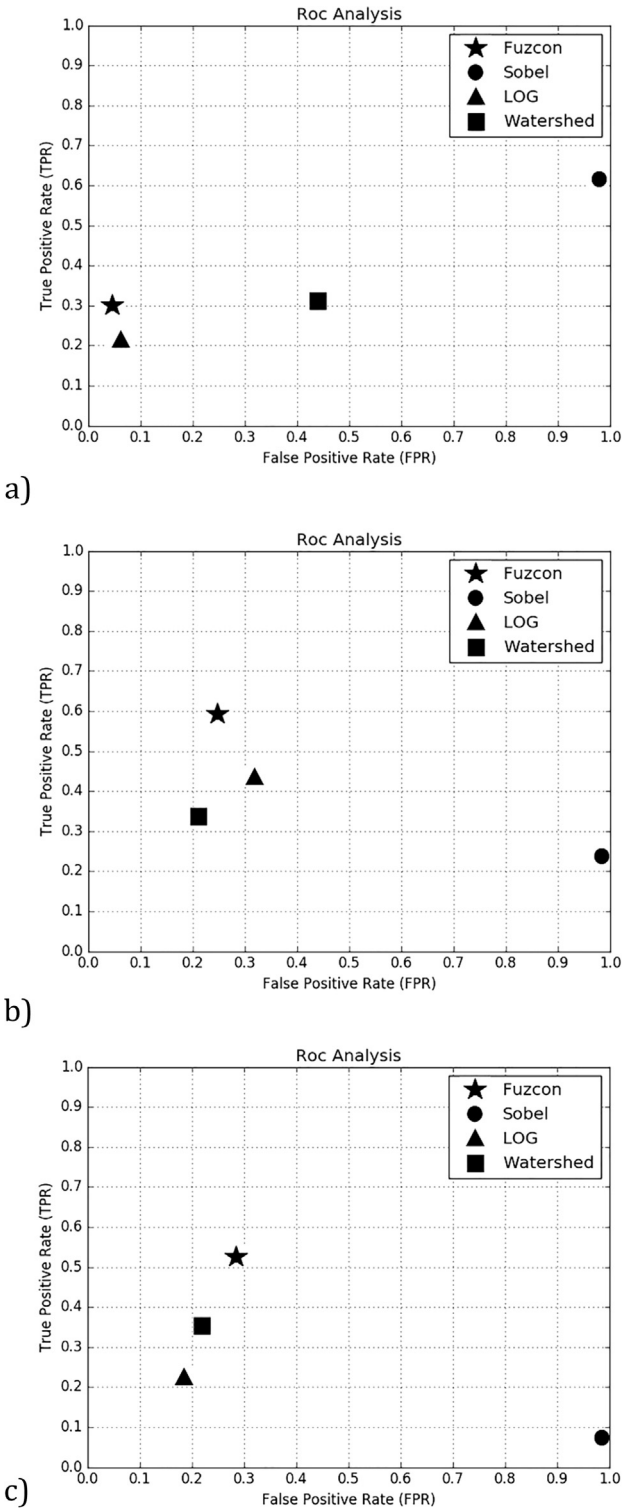


Fig. 13. ROC Analysis. a) ROC points of Fig. 10. b) ROC points of Fig. 11. c) ROC points of Fig. 12.

wider the filter matrix should be, which results in a more smoothed image, making the task of edge detection more difficult in images with high void fraction. Although it provides good results, this smoothing has destroyed some bubbles (Fig. 10c), and LOG operator could not detect all bubble edges, as shown in Figs. 11c and 12c. This limitation resulted from bubble edge thickness being larger than the filter mask (3 × 3) used.

Knowing that one of the important points of the segmentation task is

Table 2
ROC points.

Edge Detectors	Fig. 13a	Fig. 13b	Fig. 13c
Proposed (FUZCON)	TPR = 0.30	TPR = 0.59	TPR = 0.52
	FPR = 0.048	FPR = 0.25	FPR = 0.29
	TPR/FPR = 6.25	TPR/FPR = 2.36	TPR/FPR = 1.79
Sobel	TPR = 0.61	TPR = 0.24	TPR = 0.07
	FPR = 0.98	FPR = 0.99	FPR = 0.99
	TPR/FPR = 0.62	TPR/FPR = 0.24	TPR/FPR = 0.07
LoG	TRP = 0.22	TPR = 0.43	TPR = 0.22
	FPR = 0.07	FPR = 0.32	FPR = 0.19
	TPR/FPR = 3.14	TPR/FPR = 1.35	TPR/FPR = 1.16
Watershed	TRP = 0.31	TPR = 0.34	TPR = 0.36
	FPR = 0.44	FPR = 0.21	FPR = 0.22
	TPR/FPR = 0.71	TPR/FPR = 1.62	TPR/FPR = 1.64

to find finer edges, instead of the details of their nearest neighbors (Alshennawy and Aly, 2009), the proposed algorithm, differently from previous methods, considers the standard deviations for each Contrast value found. FUZCON does not use smoothing method, even when there are background noises, noises inside the bubbles, and regions with different contrast levels.

The proposed detector achieved better performance according to the ROC analysis results (Bowyer et al., 2001) shown in Fig. 13, where the higher TPR/FPR proportion represents better results due to higher TPR values associated with lower FPR values. Table 2 presents the coordinate points (TPR and FPR) and its TPR/FPR ratios of ROC's analysis illustrated in Fig. 13a–c. This algorithm has shown superior robustness to noise (Fig. 10e, 11e and 12e), as indicated by ROCs points in Fig. 13b and c, when detectors are applied to the high-density two-phase flow bubble images chosen for this work.

Although FUZCON could find a larger number of single and agglomerated bubbles, which was associated with high TPR values, and was able to subtract the background without damaging most of the continuous bubble edges, LOG operator has presented slightly better detection results (Fig. 13a) in homogeneous images where it is not possible to identify large lighting differences as in Fig. 10a. Sobel operator attained high TPR values however there was a high FPR associated, representing the detection of borders inside bubbles.

The FUZCON accuracy is strongly affected by statistical analysis performed on each gradient image. Consequently, when images include a large number of compacted bubbles, shown in Figs. 11a and 12a, identification performance is improved: the number of bubbles detected (high TP rate) with few errors (low FP rates), as shown in Fig. 13b and c.

4. Conclusions

A new approach to bubble segmentation for gas-liquid two-phase bubbly flow on natural circulation experiments, labeled FUZCON, is presented. It was applied to two-phase flow images using image processing and pattern recognition techniques.

The images were chosen from an experimental database containing 1798 images with different resolutions. An image operator *Con* was proposed based on a Contrast function definition. After image preprocessing, a Contrast Standard Deviation (CSD) methodology was developed to allow the detection of abrupt transition of the statistical distribution of evaluated CSD values through whole image. Image contrast standard deviation analysis was implemented through a self-adaptive Fuzzy Logic inference system using dynamic membership functions for each analyzed image.

Edge detection performance was evaluated using the ROC analysis method for an initial comparison with the traditionally used methods for edge detection applied to bubbles inside high-density two-phase flow images. This comparison was based on manually constructed Ground-Truth images. The results showed that the proposed detector, without the need to use smoothing algorithms and without human intervention, could improve void fraction estimation on high-density two-phase flow bubble images.

Acknowledgements

This investigation was partially supported by Brazilian Agencies Coordination for the Improvement of Higher Education (CAPES) and National Counsel of Technological and Scientific Development (CNPq). The authors would like to thank all effort on experimental setup by Dr. Luiz A. Macedo. The authors also wish to acknowledge experimental assistance from the Department of Chemical Engineering of the Polytechnic School of the University of São Paulo.

References

Andrade, D. Belchior, A. Sabundjian, G., 2000. Two-phase instabilities in a natural circulation rectangular loop. In: 8th Int. Conf. Nucl. Eng., pp. 1–5.
Mesquita, R.N. De, Masotti, P.H.F., Penha, R.M.L., Andrade, D.A., Sabundjian, G., Torres, W.M., Macedo, L.A., 2012. Classification of natural circulation two-phase flow patterns using fuzzy inference on image analysis. Nucl. Eng. Des. 250, 592–599.
Abdallah, H.A., Mohiy, M., Shaalan, A.A., 2009. SVD-based watermarking scheme in complex wavelet domain for color video. In: IEEE Int. Conf. Comput. Eng. Syst., pp. 455–460.
Crivalero, K.C.O., Seleghim, J.R., Hervieu, E., 2002. Detection horizontal of two-phase flow patterns through a neural network model. J. Braz. Soc. Mech. Sci. 6, 70–75.
Mesquita, R.N., Sabundjian, G., Andrade, D.A., Umbehaun, P.E., Torres, W.M., Conti, T. N., Macedo, L.A., 2009. Two-phase flow patterns recognition and parameters estimation through natural circulation test loop image analysis. In: Int. Conf. Boil. Heat Transf., pp. 3–7.
Sunde Carl, Avdic Senada, P.I., 2005. Classification of Two-Phase Flow Regimes via Image Analysis and a Neural-Wavelet Approach, vol. 46, pp. 348–358.
Wu, M., Dong, F., Qi, G., 2006. Feature extraction method for gas/liquid two-phase flow. Mach. Learn. 13–16.
Sarkar, P.S., Kashyap, Y., Sinha, A., Vijyan, P.K., Rao, G.S.S.P., 2005. Applications for real-time neutron radiography for convection driven flow pattern transition studies. IEEE Trans. Nucl. Sci. 52, 290–294.
Zhang, Z. Huang, T., 1988. A novel method for bubble identification based on ANN. In: 2009 Third Int. Conf. Genet. Evol. Comput, pp. 1681–1696.
Heydarian, M., Noseworthy, M.D., Kamath, M.V., Boylan, C., Poehlman, W.F.S., 2009. Optimizing the level set algorithm for detecting object edges in MR and CT images. IEEE Trans. Nucl. Sci. 56, 156–166.
Barbosa, P.R., Crivelaro, K.C.O., Seleghim Jr., P., 2010. On the application of self-organizing neural networks in gas-liquid and gas-solid flow regime identification. J. Brazil. Soc. Mech. Sci. Eng. 32, 15–20.
Dinh, T.B., Choi, T.S., 1999. Application of image processing technique, S in air/water two phase flow. Mech. Res. Commun. 26, 463–468.
Shi, L., 2007. Fuzzy recognition for gas-liquid two-phase flow pattern based on image processing. In: 2007 IEEE International Conference on Control and Automation. IEEE, pp. 1424–1427.
Shi, L, Zhou, Z. Ren, S., 2004. Parameter measurements of two-phase bubbly flow using digital image processing. In: Proc. 5th. World Congr. Intell. Control Autom., pp. 3858–3861.
Wenyin, Z.W.Z., Ningde, J.N.J., Xia, L.X.L., Zhiqiang, N.Z.N., 2008. Bubble image segmentation of gas/liquid two-phase flow based on improved canny operator. In: 2008 Int. Conf. Comput. Sci. Softw. Eng., vol. 1, pp. 799–801.
Delacroix, S., Germain, G., Gaurier, B., Billard, J.-Y., 2016. Experimental study of bubble sweep-down in wave and current circulating tank: Part II—bubble clouds characterization. Ocean Eng. 120, 88–99.
Huang, J.-Y., Kao, P.-F., Chen, Y.-S., 2007. A set of image processing algorithms for computer-aided diagnosis in nuclear medicine whole body bone scan images. IEEE Trans. Nucl. Sci. 54, 514–522.
Mishima, K., Hibiki, T., 1996. Some characteristics of air-water two-phase flow in small diameter vertical tubes. Int. J. Multiphase. Flow 22, 703–712.
Wang, H., Dong, F., 2009. Image features extraction of gas/liquid two-phase flow in horizontal pipeline by GLCM and GLGCM. In: 2009 9th Int. Conf. Electron. Meas. Instruments 2-135-2-139.
Do Amaral, C.E.F., Alves, R.F., da Silva, M.J., Arruda, L.V.R., Dorini, L., Morales, R.E.M., Pipa, D.R., 2013. Image processing techniques for high-speed videometry in horizontal two-phase slug flows. Flow Meas. Instrum. 33, 257–264.
Lau, Y.M., Deen, N.G., Kuipers, J.A.M., 2013. Development of an image measurement technique for size distribution in dense bubbly flows. Chem. Eng. Sci. 94, 20–29.
Barkhoda, W., Tab, F.A., Shahryari, O.K., 2009. Fuzzy edge detection based on pixel's gradient and standard deviation values. In: Proc. Int. Multiconference Comput. Sci. Inf. Technol. IMCSIT '09 4, pp. 7–10.
Patel, J. Patwardhan, J. Sankhe, K. Kumbhare, R., 2011. Fuzzy inference based edge detection system using sobel and laplacian of gaussian operators. In: Int. Conf. and Workshop on Emerg. Trends in Tech (ICWET).
Jzau-Sheng, L., Cheng, Kuo-Sheng, Mao, Chi-Wu, 1996. A fuzzy Hopfield neural network for medical image segmentation. IEEE Trans. Nucl. Sci. 43, 2389–2398.
Bhardwaj, S., Mittal, A., 2012. A survey on various edge detector techniques. Procedia

Technol. 4, 220–226.
Miché, P., Debrie, R., 1995. Fast and self-adaptive image segmentation using extended declivity. Ann. Des Télécommun. 50, 401–410.
Bensrhair, A., Miché, P., Debrie, R., 1996. Fast and automatic stereo vision matching algorithm based on dynamic programming method. Pattern Recognit. Lett. 5, 457–466.
El Ansari, M., Mousset, S., Bensrhair, A., 2010. Temporal consistent real-time stereo for intelligent vehicles. Pattern Recognit. Lett. 31, 1226–1238.
Cabani, I, Toulminet, G., Bensrhair, A., 2006. A color stereo vision system for extraction of 3D Edges of obstacle. In: Proceedings of the 2006 IEEE Intelligent Transportation Systems Conference (ITSC) 17, pp. 307–312.
Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE Trans. Sys. Man. Cyber 9 (62–66), 1979.
Gonzales, Raphael C., Woods, R.E., 2007. Digital Image Processing. Prentice Hall.
Jayachandran, A., Dhanashakeran, R., Anand, O.S., Ajitha, J.H.M., 2010. Fuzzy information system based digital image segmentation by edge detection. In: 2010 IEEE Int. Conf. Comput. Intell. Comput. Res., pp. 1–5.
Heath, M., Sarkar, S., Sanocki, T., Bowyer, K., 1998. Comparison of edge detectors. Comput. Vis. Image Underst. 69, 38–54.
Bowyer, K., Kranenburg, C., Dougherty, S., 2001. Edge detector evaluation using empirical ROC curves. Comput. Vis. Image Underst. 84, 77–103.
Meyer, F., 1991. Un algorithme optimal pour la ligne de partage des eaux. Dans. In: 8. Congrès de Reconnaissance Des Formes et Intelligence Artificielle, pp. 847–857.
Sobel, I., 1974. On calibrating computer controlled cameras for perceiving 3-D scenes. Artif. Intell. 4, 185–198.
Marr, D., Hildreth, E., 1980. Theory of edge detection. Proc R. Soc London 207, 187–217.
Alshennawy, A., Aly, A., 2009. Edge detection in digital images using fuzzy logic technique. World Acad. Sci. Eng. 51, 185–193.