

Received December 20, 2021, accepted January 19, 2022, date of publication January 27, 2022, date of current version February 10, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3147206

# A Robust SVM Color-Based Food Segmentation Algorithm for the Production Process of a Traditional Carasau Bread

KATIUSCIA MANNARO<sup>1</sup>, MATTEO BAIRE<sup>2</sup>, ALESSANDRO FANTI<sup>2</sup>, (Member, IEEE),  
MATTEO BRUNO LODI<sup>2</sup>, (Graduate Student Member, IEEE), LUCA DIDACI<sup>2</sup>, (Member, IEEE),  
ALESSANDRO FEDELI<sup>3</sup>, (Member, IEEE), LUISANNA COCCO<sup>1</sup>,  
ANDREA RANDAZZO<sup>3</sup>, (Senior Member, IEEE), GIUSEPPE MAZZARELLA<sup>2</sup>, (Senior Member, IEEE),  
AND GIORGIO FUMERA<sup>2</sup>, (Member, IEEE)

<sup>1</sup>Department of Mathematics and Computer Science, University of Cagliari, 09123 Cagliari, Italy

<sup>2</sup>Department of Electrical and Electronic Engineering, University of Cagliari, 09123 Cagliari, Italy

<sup>3</sup>Department of Electrical, Electronic Telecommunications Engineering, University of Genoa, 16145 Genoa, Italy

Corresponding authors: Katuscia Mannaro (katuscia.mannaro@unica.it) and Alessandro Fanti (alessandro.fanti@unica.it)

This work was supported in part by the Project Ingegnerizzazione e Automazione del Processo di Produzione Tradizionale del Pane Carasau mediante l'utilizzo di tecnologie IoT (IAPC) funded by the Ministero dello Sviluppo Economico in AGRIFOOD PON I&C (2014–2020) under Grant CUP: B21B19000640008 COR: 1406652.

**ABSTRACT** In this paper, we address the problem of automatic image segmentation methods applied to the partial automation of the production process of a traditional Sardinian flatbread called pane Carasau for assuring quality control. The study focuses on one of the most critical activities for obtaining an efficient degree of automation: the estimation of the size and shape of the bread sheets during the production phase, to study the shape variations undergone by the sheet depending on some environmental and production variables. The knowledge can thus be used to create a system capable of predicting the quality of the shape of the dough produced and empower the production process. We implemented an image acquisition system and created an efficient machine learning algorithm, based on support vector machines, for the segmentation and estimation of image measurements for Carasau bread. Experiments demonstrated that the method can successfully achieve accurate segmentation of bread sheets images, ensuring that the dimensions extracted are representative of the sheets coming from the production process. The algorithm proved to be fast and accurate in estimating the size of the bread sheets in various scenarios that occurred over a year of acquisitions. The maximum error committed by the algorithm is equal to the 2.2% of the pixel size in the worst scenario and to 1.2% elsewhere.

## INDEX TERMS

Computer vision, food technology, image color analysis, image processing, image segmentation, machine learning, support vector machines.

## I. INTRODUCTION

*Pane Carasau*, also known as “musical sheet” (in Italian “carta da musica”) given the sound it makes when it is broken and chewed, is a traditional Sardinian flatbread [1]. It is an Italian excellence with a limited production, linked to a territory and its history. The term derives from the Sardinian verb “carasare” which is the second baking phase to complete the process and to obtain the finished product.

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

It consists in large and thin crunchy sheets of bread, without crumbs, with a discoidal shape, a variable diameter (between 18 and 45 cm), and 0.7 to 1.0 mm thick depending on the place of production. It is obtained from the processing of high quality durum wheat remilled semolina, natural yeast, sea salt and dechlorinated tap water. Carasau is classified as Italian Traditional Agrifood Product (in Italian PAT), that is an official approval for traditional Italian regional food products similar to the Protected Geographical Status of the European Union. A PAT requires to be obtained with processing, preservation and seasoning methods that are

consolidated over time and homogeneous within the entire territory concerned according to traditional rules [2]. Mass production of food products has resulted in great increase in the efficiency of food producing plants. Quality control and assurance mechanisms are needed in the production lines of processing and manufacturing plants [3]. In particular, during the phases of the Pane Carasau production process, the bread sheets undergo changes in shape as a function of some environmental variables such as temperature and humidity of the environment, but also depends on mechanical variables, such as the speed of the cutting organs and conveyor belts detected during the process [4]. The demand for the product has been constantly increasing, so traditional bakeries, in order to meet the needs of the market supply, while maintaining high product quality standards, must rationalise their plants with particular reference to certain phases of production. Nowadays, food quality control and assurance in the producing lines may be supported by recent developments in automatic image segmentation and supporting technologies.

According to Cheng et al. [5], image segmentation is one of the most critical and essential tasks in image processing and its accuracy determines the quality of the final result of analysis, i.e., the eventual success or failure of computerized analysis procedure. Briefly, image segmentation is a computer vision task which consists in segmenting a given image into different regions, according to criteria such as high intra- and low inter-homogeneity between regions, to distinguish and separate the different elements of an image (i.e., foreground and background). This task is of interest in a variety of fields such as medical image processing [6], object detection [7], biometric recognition systems [8], video surveillance [9] and recently also in agrifood sector [10], [11]. Despite many image segmentation algorithms have been proposed so far, this is still a challenging research topic in computer vision. Indeed, each specific application often requires the adaptation of existing solutions to its peculiarities. This turned out to be the case of the application considered in this paper as well.

This study has two main goals.

Firstly, the investigation of the possibilities of implementing image acquisition system and a computer vision algorithm for estimating the geometric parameters (major and minor axis) of the *Carasau* bread sheets at the entrance of the leavening room on a uniform conveyor belt, and at the exit of the leavening room on grid conveyor belt for assuring the effectiveness and quality in the production line of Carasau bread. The evaluation of the size of the bread sheets at the entrance and exit of this room will be used to automatically estimate and predict any corrections and process parameters immediately after cutting the sheet based on the correlation with the environmental variables (temperature and humidity) and production variables (speed of the cutting devices and conveyor belts, etc.) estimated with other methods [12]. Secondly, developing a system capable of predicting the quality of the shape of the dough produced and therefore can

be used to optimize and control its final shape. For this to be possible, an accurate images segmentation of the bread sheets is required to ensure that the dimensions extracted from the images are representative of the sheets coming from the production process. This system might be generalized and later applied in more complicated pattern recognition applications.

The rest of this paper is organized as follows. Section II presents an overview on the production process. Works related to segmentation and measure extraction approaches are reviewed in Section III. Section IV contains the description of our proposed solution. The details of the segmentation algorithm are presented in Section V. Section VI shows the experimental results and then the performances of our solution. Finally, Section VII concludes the paper and suggests future research directions.

## II. CRITICAL ASPECTS AND ISSUES ABOUT THE CARASAU BREAD PRODUCTION

All steps in bread processing are important for a successful operation and to obtain a quality final product, but the four truly vital process steps for the production of Pane Carasau are: i) preparation of the dough (kneading) made from twice-milled durum wheat flour, water, salt and yeast and proofing; ii) shaping and cutting of the disks of pane carasau; iii) baking and separation by hand of the two layers obtained; iv) “*carasatura*” - the stage that gives its name to the flatbread - or second baking. The bread must be produced in Sardinia with mechanical and physical processes, which aim to guarantee the best organoleptic quality to the product. In addition, it has to be produced in structures suitable for guaranteeing adequate hygienic-sanitary conditions [2]. The final product must be packaged using food containers produced according to the regulations in force, hence sealed packages constituting a physical barrier impermeable to the atmospheric-physical and polluting agents, and must be suitably labeled. For a better understanding of the Carasau bread production we show in Fig. 1 an example scheme as reported in Baire et al. [12].

Following the illustrative scheme of the bread production phases, Fig. 1 shows that (1) raw materials are kneaded to produce the dough, which, after a first leavening (2) is shaped, sheeted (3) and cut into disks (4). Then, following a second leavening in a dedicated room (5), the disks are baked once (6), controlled by operators, re-cut and separated to obtain two sheets (7). Finally, after a second baking (8), the flat breads are weighted. In this figure S1 and S2 indicate the sensors collocated in the dough room and in the leavening room for the monitoring of environmental parameters (temperature and relative humidity in air, and concentrations of CO and CO<sub>2</sub>). With E1, E2 and E3, we indicate the encoders for the measurements of the cinematic parameters of the belt. Finally, for the case of our interest, we visualize the position of C1 and C2 that are two optical cameras collocated at the entrance and at the exit of the leavening room for the image processing system that is for

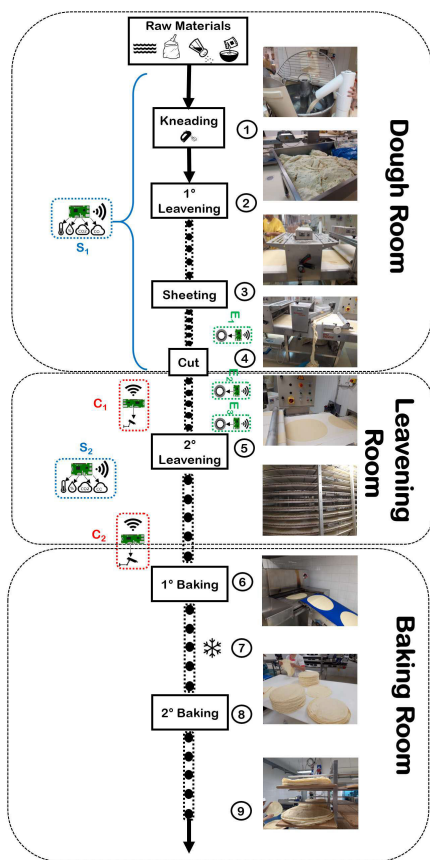


FIGURE 1. Illustrative scheme of the stages of production of the Pane Carasau [12].

the optical monitoring of bread’s morphological parameters. Indeed, in this research work we analyze a specific part of the production process that concerns the phase in which the conveyor belt carries the bread sheets into the leavening room. In the leavening room the bread sheets move along the conveyor for the whole leavening process which lasts about 20-30 minutes [12]. At the end of the leavening process, these discs come out of the leavening room and are directed to a toasting machine. During leavening the bread sheets undergo deformation. To overcome this problem, we tried to develop a system capable of predicting the quality of the shape of the dough produced and therefore can be used to optimize its final shape.

An accurate images segmentation of the bread sheets is required to ensure that the dimensions extracted from the images are representative of the sheets coming from the production process. In order to develop a new segmentation method, an in-depth analysis of the state of art must be performed.

To this aim, no “off-the-shelf” solution turned out to be available in the literature. This motivated us to develop an ad hoc image processing pipeline involving a first phase for image pre-processing and feature extraction, a second one for image segmentation based on the Support Vector Machine (SVM) classifier, and finally the extraction of

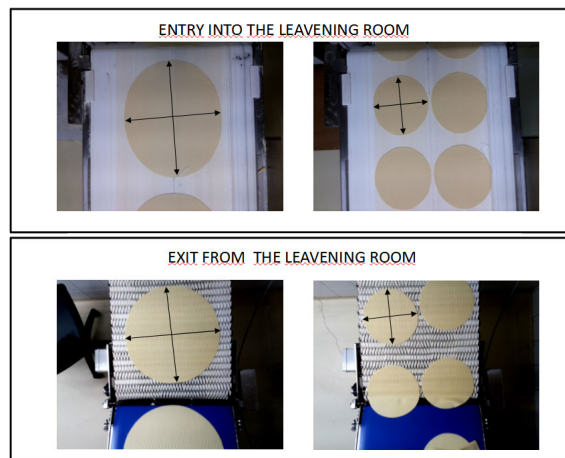


FIGURE 2. On the left the bread sheets with a diameter of 45 cm and on the right the bread sheets with a diameter of 18 cm.

the dough shape and dimensions from the outcome of the segmentation step.

### III. RELATED WORK

There is a vast body of work in the areas of foreground segmentation and machine vision systems for size estimation. Here we briefly introduce the major image segmentation approaches in the literature: thresholding, template matching, seeded region growing, edge detection, artificial neural networks, clustering, watershed and image classifiers, e.g., SVMs. All these techniques have been proposed and evaluated either for generic images, or for specific applications. In both cases, usually they cannot be directly used in novel, specific applications, but require some kind of adaptation.

The accuracy of size estimation methods depends on the accuracy of the segmentation and detection methods [13]. Furthermore, the research works in the literature that propose new segmentation approaches generally do not deal also with estimating the dimensions of segmented objects, whereas the works that deal with dimension estimation generally perform under controlled scenarios, or assume that segmentation can be done easily or with the use of basic algorithms. A general discussion of the major segmentation algorithms is reported below.

**Thresholding** is one of the most common and simplest approach to segment an image. Pixels are partitioned depending on whether their intensity value is below a given threshold or not. The key parameter is the choice of the threshold value. Thresholding techniques can be classified into two categories: Global Thresholding, subdivided in turn into Traditional, Iterative and Multistage, and Local Thresholding [14]. Global Thresholding methods are used when the intensity distribution of foreground and background objects are very distinct and it is possible to separate them using one threshold on the entire image. Otsu’s algorithm is the most popular global traditional thresholding technique [15]; it aims at finding the optimal value for the global threshold. In comparison with global techniques, local

thresholding ones, also known as adaptive thresholding, use different threshold values for different regions in the image. They perform better in presence of noise, e.g., when dealing with information near texts or objects. The thresholding approach has received wide attention because of its low complexity, low storage requirements and low processing time [16]. However, it is not effective if the different regions in the image under consideration have not well-defined areas (e.g., different objects with similar gray areas), or the pixel intensity histograms are unimodal [17]. For detailed descriptions of recent thresholding segmentation approaches, readers may refer to [17]–[20].

**Template Matching** is a digital image processing technique with the aim of finding the optimal location and orientation of a template image in the input one, often as part of a larger problem of object recognition [21], [22]. It consists in calculating at each position of the input image a function that measures the degree of similarity between a template and a portion of it. Template matching is used in applications such as face recognition, signal processing, video compression, and medical image processing [23]–[26]. It is efficient for small objects, but requires a high processing time for complex patterns and large images. Some advantages and disadvantages of segmentation methods based on template matching are discussed in [27].

**Seeded region growing (SRG)** algorithm is an attractive region-based method for segmenting intensity images proposed by Adams and Bischof [28] in 1994. This approach to segmentation is also classified as a pixel-based image segmentation method since it assembles pixels into larger regions based on predefined seed pixels, growing criteria, and stop conditions by determining whether the pixel neighbors should be added to the region [29]. The algorithm is characterized by fast execution, robust segmentation, and no tuning parameters, but is inherently dependent on the order of pixel processing [30]. It is a sequential technique, and the results depend on the order with which the pixels are examined for labeling and automatic seed selection, thus leading to the risks of losing the global vision of the problem. It is very demanding in terms of time and memory requirements [31], [32]. In Shih and Cheng [33] an efficient segmentation algorithm is presented for color images with automatic seed selection and also developed strategies to avoid order dependencies.

**Edge detection** is an image processing technique used to find the boundaries of objects within digital images by identifying points with discontinuities in brightness. These points, where the image brightness varies sharply, are called edges (or boundaries) of the image, so an edge is a boundary between two homogeneous regions [34]–[37]. These algorithms are particularly suitable in contexts with a high contrast between subject and background. It is important to choose suitable edge detectors to get the best results from the matching process [38]. Unfortunately, scarce results are obtained in the presence of many edges, closed curves or corners where the gray level intensity function varies.

Furthermore, some edge detection algorithms are noise sensitive, and if the image intensity varies gradually they do not work at all [39], [40].

**Artificial neural networks (ANNs)** (or simply neural networks) have been successfully used for food control and product classification [41]. A neural network is a set of units (artificial neurons) with a predefined pattern of weighted connections (e.g., feed-forward multi-layer), that is inspired by the structure of the human brain. Their flexibility allow them to deal with highly non-linear problems. ANNs can learn to recognize patterns in image, video, sound, text data, etc. This input data, represented as feature vectors, goes through all the layers, as the output of one layer is fed into the next layer, and they automatically “learn” identifying characteristics from the examples that they process. ANNs are very effective pattern classifiers because of their ability to learn highly non-linear input-output relationships, and to deal with uncertainty, noise and randomness [42]. A specific class of ANNs commonly used in image processing is the convolutional neural network (CNN, or ConvNet) belonging to the class of deep NNs [41]. A CNN is a multilayer neural network made up of convolutional layers and pooling layers, whose neurons take small patches of the previous layer as input, and fully connected layers. CNNs are currently considered as one of the most popular machine intelligence model for big data analysis in various research areas; a number of CNN architectures with feedback mechanisms applied to image classification and image recognition have been proposed in literature [43]–[47].

ANNs have been applied in almost every aspect of food science for modeling tasks in process control and simulation and also for food safety and quality control. Özsert Yiğit and Özyildirim [48] proposed three CNN structures trained from scratch by using different learning methods and compared its performance with pre-trained structures. Alexnet and Caffenet fine-tuned with the same learning algorithms for developing a pre-trained model for food recognition. Image recognition of food items is generally very difficult. CNNs are a state-of-the-art approach to deep learning that has been shown recently to be very effective in this application.

Kagaya et al. [41] applied a CNN for food image recognition and detection through parameter optimization. They built a food image data set of the most frequent food items uploaded by a large number of real users and used it to evaluate recognition performance. CNNs showed significantly higher accuracy than traditional SVM-based methods with handcrafted features. They found that the convolution kernels show that color dominates the feature extraction process. The findings in Kamilaris and Prenafeta-Boldú [49] indicate that CNNs have a high precision in the large majority of the problems where they have been used, attaining a higher precision than other popular image-processing techniques. A main advantage is the ability to approximate highly complex problems effectively. On the other hand, a main disadvantage is that CNNs sometimes take much longer to train though after training, their testing time

is much lower than other methods such as SVMs. Another important disadvantage is the need for large data sets, since a manual annotation must be performed by domain experts.

**Clustering** is one of the most popular approaches to unsupervised image segmentation. It is based on pixel classification, and consists in subdividing an image into regions made up of clusters of pixels in feature space that exhibit a high intra-cluster and low inter-cluster similarity [50]. When data is represented by a small number of clusters, necessarily certain fine-grained details are lost in favor of problem simplification [51]. A large quantity of research studies have recently applied the correlation clustering framework to image segmentation, and there was a lot of effort to extract better region-based features between neighboring superpixels [52]–[55]. Zhou and Wei [56] proposed an unsupervised segmentation framework based on a novel deep image clustering (DIC) model and the results showed that DIC is less affected by the segmentation parameter, such as cluster numbers, and of lower computation cost. The  $K$ -means algorithm (or Lloyd's algorithm) is the most commonly used technique in the clustering-based segmentation field. It is one of the key techniques in pixel-based methods [57]. Significant improvements on the final segmentation results may be achieved by using notably more sophisticated feature selection procedures, more elaborate clustering techniques, such as mixture of different or non-Gaussian distributions for the multidimensional texture features and taking into account prior distribution on the labels and region processes [58]. The  $K$ -means clustering algorithm is simple to implement and robust, its computational complexity is relatively low compared with other region-based or edge-based methods, and it provides comparatively good results if clusters in data sets are distinct or well separated into clusters, therefore its application is more practicable [57], [59]. Mignotte [58] presented a new segmentation strategy based on a fusion procedure whose goal was to combine several segmentation maps in order to obtain a more reliable and accurate segmentation result. This framework is simple, fast, easily parallelizable, general enough to be applied to various computer vision applications. In [57] the authors proposed a method that combines histogram statistics and  $K$ -means clustering to track the tumor objects in magnetic resonance (MR) brain images. They converted a given gray-level magnetic resonance image into a color space image and then separated the position of tumor objects from other items of an MR image. Their preliminary experiments demonstrated encouraging results.  $K$ -means has been used for performing food image segmentation and Fuzzy C-Means is most popular clustering method similar to  $K$ -Means that uses fuzzy theory to improve clustering results and become an important tool in many applications [60]. Dubey et al. [61] presented a novel defect segmentation of fruits by using color images of fruits (they took the apples as a case study) and  $K$ -means clustering unsupervised algorithm. They provided a feasible robust solution for defecting segmentation of fruits. The experimental results showed the effectiveness

of proposed approach to improve the defect segmentation quality in aspects of precision and computational time. Zheng et al. [62] proposed an adaptive  $K$ -means image segmentation method, which generated accurate segmentation results with simple operation and achieved good results in the field of food image. Food segmentation is not easily performed if the image has low contrast with its background or the background is not homogeneous. In order to overcome this problem, Siswanto et al. [63] proposed Sobel operator for determining the region of interest combined with  $k$ -means clustering for separating object and background in the region of interest.

**Watershed** segmentation algorithm can be classified as a region-based segmentation approach. This popular image segmentation technique is based on the representation of a gray-scale image in the form of a topographic relief [64] consisting of low-lying valleys (minimums), high-altitude ridges (watershed lines) and slopes (catchment basins), which is flooded by water. Practically, it exploits a three-dimensional interpretation of the image intensity gradient to create reliefs that separate the image into various basins that can be "flooded" until the boundaries of the segments where the flooding stops thanks to the creation of appropriate "dams". A critical review of several definitions of the watershed transform, both for the continuous and the discrete case, and the associated sequential algorithms was presented in [65]. Kornilov and Safonov [66] provided a list of software for watershed segmentation and an interesting analysis of the limitations for processing of huge images. Watershed methods differ mainly in their approaches of placing the seeds and defining the priorities. In [67] the authors proposed a watershed-based superpixel algorithm that preserves the color homogeneity with global and local boundary marching. Their method searches for boundaries using the mean color information of the super-pixels, and defines the region content in the gradient information. Watershed-based methods typically run at high speed, they are simple and intuitive, widely utilized in dealing with complex background images, provide more stable results and the contours obtained are continuous, finally they are able to produce a complete division of the image in separated regions if the contrast is poor [68], [69]. It can be easily adapted to any kind of digital grid and extended to  $n$ -dimensional images and graphs and should allow anyone to resort to watersheds for solving complex segmentation problems [64]. On the other hand in watershed methods calculating gradients is complex, their drawbacks include over-segmentation and sensitivity to noise [68] and they have lower accuracies than clustering-based methods [67].

**SVMs** are a machine learning method introduced by Cortes and Vapnik [70] in the early 1990's for classification and regression problems, based on the *structural risk* minimization principle. It has been successfully applied to numerous pattern recognition problems. For two-class classification problems, SVMs determine the hyperplane that separates with maximum margin the training examples of the

**TABLE 1. Comparison of some segmentation techniques.**

Method	Cons	Reference
<ul style="list-style-type: none"> <li>• A pixel-based color image segmentation using SVM (classifier) and Fuzzy C-means.</li> <li>• Extraction of the pixel-level color and texture features via the local spatial similarity measure model and steerable filter.</li> <li>• SVM model is trained by using fuzzy C-means with the extracted pixel-level features.</li> </ul>	<ul style="list-style-type: none"> <li>• General purpose</li> <li>• Random training sample selection and Fuzzy C-means training sample selection.</li> <li>• No prior knowledge about the problem.</li> </ul>	[75]
<ul style="list-style-type: none"> <li>• Usage of two competing one-class support vector machines at each pixel location to model local color distributions for foreground and background.</li> <li>• It captures the local foreground and background color densities separately.</li> <li>• Determination of a proper label for the pixel jointly.</li> <li>• The algorithm propagates initial user labeling to the entire image by iterating between training local SVM and applying them to label the pixels.</li> </ul>	<ul style="list-style-type: none"> <li>• A training set is not used.</li> <li>• General purpose.</li> <li>• A priori knowledge is not exploited.</li> </ul>	[72]
<ul style="list-style-type: none"> <li>• Goal: to locate and segment the line defects in X-ray images.</li> <li>• SVM technique to classify the defect and non-defect features to obtain a coarse defect region.</li> <li>• Area extraction with an adaptive segmentation algorithm.</li> <li>• Features of defects obtained from the extracted weld area by multiple thresholds.</li> <li>• SVM training with extracted features.</li> </ul>	<ul style="list-style-type: none"> <li>• Black &amp; White images</li> </ul>	[73]
<ul style="list-style-type: none"> <li>• Goal: development of a cascaded two-stage SVM-based classifier to determine the wound boundaries.</li> <li>• A linear iterative clustering (SLIC) method is used to segment the image into a number of superpixels</li> <li>• SVM training through Fuzzy c-means on extracted feature.</li> <li>• Badly classified pixel collection.</li> <li>• Extraction of new features and second SVM training.</li> <li>• Detected wound boundary is refined by applying a conditional random field method.</li> </ul>	<ul style="list-style-type: none"> <li>• Recall adequate for the user application, but less than approximately 80%</li> <li>• Slow segmentation (&gt; 20 s).</li> </ul>	[74]

two classes in feature space, i.e., the one that maintains the maximum possible distance from the closest points of each class [71]. Interestingly, their efficiency does not directly depend on the feature dimension of classified entities. SVMs have been extensively studied and have been successfully used in many applications. Gong et al. [72] used single-class SVMs for real-time foreground and background separation in videos. Wang et al. [73] used them to classify the defect and non-defect features, to obtain a coarse defect region. Wang et al. [74] developed a cascaded two-stage SVM-based classifier to determine the wound boundaries on foot ulcer images. SVMs can also be used to solve color image segmentation problems [75]. For instance, Wang et al. [75] presented an efficient pixel-based color image segmentation method using SVMs and Fuzzy C-Means.

Years of research in segmentation have demonstrated relevant improvements on the final segmentation results by using optimization and clustering techniques. As for segmentation algorithms, we should bear in mind that it is difficult for generic segmentation algorithms or, conversely, specific segmentation algorithms for a given application, to work well in other specific applications. In our case we are faced, on one hand, with a relatively simple segmentation problem with an object of almost uniform color on an almost uniform background, on the other hand we have a

non-uniform case with a non-regular pattern, the case of the exit from the non-blue grid. We propose in this paper an approach that explores and combines new segmentation models in order to get final reliable segmentation results. As a first option, we tried to segment and classify the individual pixels of the bread sheet or background based on their color among the RGB (red, green and blue) values. Since it is not easy to manually define a classification rule [76], we have opted to use a machine learning-based approach that requires the manual segmentation of a certain number of images to build a training set. Our study has reviewed a large quantity of research papers in several segmentation area. In this section we have given a glimpse of some state-of-the-art segmentation models and methods to estimate image dimensions in order to evaluate what approaches are currently in use that could be of interest to the case study. The most recent work in this area inherent to our work has been summarized in Tables 1 and 2 based on our two main objectives, i.e segmentation and dimensions extraction, on the main idea related to the method followed by the authors and the cons of the proposed approach in relation to the needs specified in our specific case.

A good segmentation algorithm for our specific case must satisfy the following requirements: i) a specific solution for the case in question that takes into account the color

**TABLE 2. Comparison of some measure extraction techniques.**

Method	Cons	Reference
<ul style="list-style-type: none"> <li>Search all edge pixels of the outer contour.</li> <li>Pixels detection via the minimum and the maximum x and y coordinates.</li> <li>Least square circle method to define a primary outer circle that fit the extracted four pixels and to calculate its center and radius.</li> </ul>	<ul style="list-style-type: none"> <li>White background image (simple segmentation)</li> <li>Method suitable for circular images or images that can be inscribed in a circle.</li> <li>A root diameter algorithm is used to calculate the root diameter from the extracted edge pixels.</li> </ul>	[77]
<ul style="list-style-type: none"> <li>A user-coded ImageJ machine vision analysis plugin to determine the orthogonal dimensions of food grains with accuracy (&gt;96.6% overall) and speed (<math>254 \pm 125</math> particles/s).</li> <li>Length and width dimensions determined from the fitted-ellipses' centroid coordinates and major axis inclination using the developed pixel-march.</li> </ul>	<ul style="list-style-type: none"> <li>A machine vision ImageJ plugin developed in Java for orthogonal length and width determination.</li> </ul>	[78]
<ul style="list-style-type: none"> <li>A machine vision system designed to measure the fruit diameter of the target berry.</li> <li>Medium filtering and bilateral filtering are performed respectively, to reduce noises and to preserve edges.</li> <li>Otsu algorithm implemented for segmenting the target berry.</li> <li>Determination of the candidate diameters by varying the angle from a minimum to a maximum.</li> <li>The position of the fruit diameter is the one with the minimum diameter.</li> </ul>	<ul style="list-style-type: none"> <li>A good segmentation but obtained with a thresholding method</li> <li>The diameter direction deviation is over a range of a small angle.</li> </ul>	[79]
<ul style="list-style-type: none"> <li>Combination of two methods (pixel size and 3D coordinates)</li> <li>Euclidean distances between two individual pixels in an apple region were determined using 3D pixel coordinates.</li> <li>The major axes of the apples were estimated using: 3D coordinates of apple regions and pixel size within apple regions.</li> <li>The sizes of all pixels along the major axis were estimated using a regression model.</li> </ul>	<ul style="list-style-type: none"> <li>Otsu's thresholding method applied to create binary images by separating image pixels into foreground and background classes.</li> <li>Use of high resolution 3D cameras is required</li> </ul>	[13]

information and that is also suitable if the background is not uniform; ii) the segmentation must be fast in order to evaluate all the bread sheets; iii) it must have high accuracy and recall; iv) it must be adaptable to the case of a change of background and able to measure the dimensions of the subject (disk with diameter of 18 cm and disk with diameter of 36 cm).

#### IV. OVERVIEW OF THE PROPOSED SOLUTION

In Figure 3 we graphically illustrate the proposed solution. The acquired image is first cropped, then in the segmentation phase it is transformed into a black and white (BW) image in which the white pixels are those that the SVM has associated with the bread while the black pixels are those that the SVM has classified as background (conveyor belt or grid as appropriate). Subsequently, the dimension extraction block picks up the contours with the Canny Edge Detection [80] and then Suzuki Border Following [81] methods. Finally, the measurements are extracted with the minimum bounding rectangle (MBR), thus approximating the dimensions of the bread sheet with those of the minimum bounding rectangle that encloses the contours of the bread detected.

The detail of each processing step is described below.

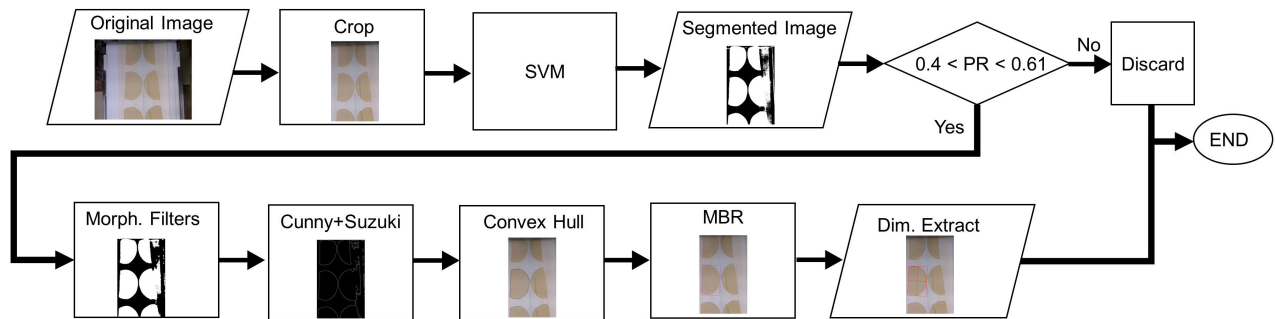
##### A. IMAGE ACQUISITION SYSTEM

The video acquisition system consists of two optical cameras (C1 and C2 in Fig. 1 collocated at the entrance and at

the exit of the leavening room) mounted on Raspberry Pi3 Model B+ [12]. Software for image acquisition, software for image processing and estimation of geometric parameters and finally software for sending comparison images to a specific Dropbox folder are installed on this processor. The software is written in Python 3. The image acquisition software performs two main functions: i) to view the live streaming, on the local network, of the bread transport process in and out of the leavening room; ii) to acquire images of the inlet and outlet dough with a resolution of  $1024 \times 768$ . These two functions cannot be performed separately, as Raspberry allows access to the camera to only one application at a time. For this reason the Python script starts the streaming through a separate thread before starting the image acquisition process obtaining the digital image from the cameras. In order to view the streaming, the software interfaces with FFmpeg,<sup>1</sup> a suite that allows to record, convert and play audio and video. Image acquisition process was timed to take five consecutive photos every second and wait one minute before taking the next five. We used picamera<sup>2</sup> package as interface to the Raspberry Pi camera module for Python 3.2 (or above). The streaming is configured to be displayed via HTTP at the local address (localhost). A fixed IP has been assigned to the two Raspberry

<sup>1</sup><https://www.ffmpeg.org/>

<sup>2</sup><https://picamera.readthedocs.io/en/release-1.13/>

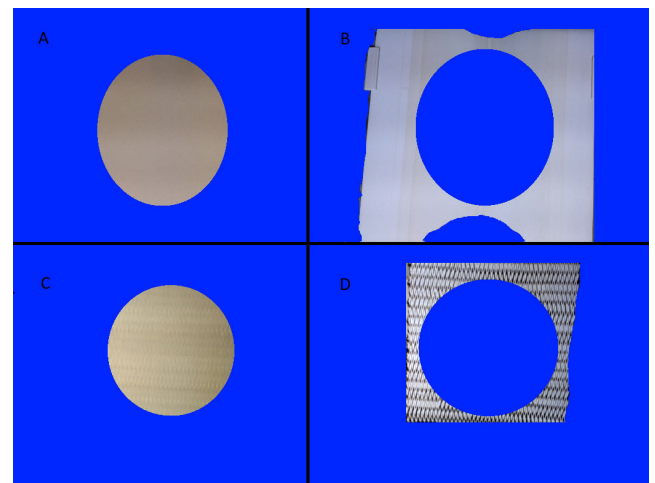


**FIGURE 3.** The proposed solution: the original image is cropped, then the Support Vector Machine (SVM), is applied to segment the image. The pixel ratio (PR) is evaluated according to two empirical thresholds. If the PR is within the correct range, morphological filters are applied, then the Cunny and Suzuki algorithms are used. The convex hull is derived and the minimum bounding rectangle (MBR) is computed. Finally, the sizes are extracted.

that manage the camera, in order to make the streaming easily accessible by the oven staff that can view the streaming on any device (PC, Tablet, etc.) connected to the local network. The images processed by the segmentation and measurement estimation algorithm were saved and collected in a Dropbox folder. The parameters estimated by the algorithm are: i) the major and minor axes expressed both in pixels and in mm; ii) the eccentricity; iii) the ratio between the pixels associated with the bread and the total pixels; iv) the ratio between the minimum area bounding rectangle and the total area of the image. All these data are saved in Elasticsearch<sup>3</sup> database. Algorithm performance monitoring via images was carried out from January 2020 until March 2021.

### B. IMAGE ANALYSIS OF PANE CARASAU

To establish the most suitable method for estimating the parameters of the bread sheet, a sample of photos taken by the two cameras described above was acquired. Before being analyzed or processed by an algorithm for estimating the values, the images need a pre-processing phase that eliminates useless parts and reduces their size. In this way, the algorithm can process only the pixels that can contain the sheet to be measured, which reduces processing time. Image analysis consists of visualizing the image pixels in the RGB space. In order to more easily distinguish the pixels belonging to the sheet from those belonging to the background, they have been manually separated by 10 images by type (input or output). Figure 4 shows the result of this operation on two images, from which an image containing only the dough bread and an image containing only the background are obtained. The excess parts (elements external to the conveyor belt or the grid) have been removed and colored in blue so that they do not overlap the regions being analyzed. Starting



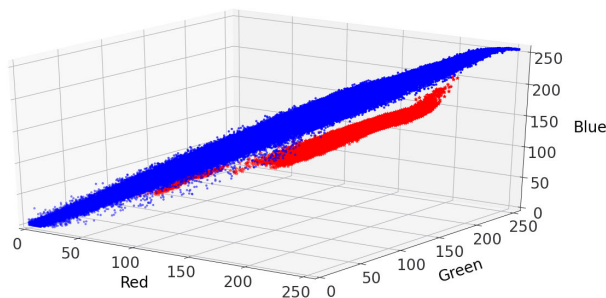
**FIGURE 4.** Example of images used to display the pixels of the bread and the grid in the RGB plane. A) Bread sheet entering the leavening cell. B) Conveyor belt that transports the bread to the entrance to the leavening cell. C) The same bread sheet at the exit of the leavening cell. A slight deformation is evident. D) Final part of the spiral cooler conveyor belt leaving the leavening cell.

from the images processed in this way, it is easier, during the viewing phase in the RGB plane, to distinguish the two elements of interest, automatically associating different colors to the pixels of the two different elements.

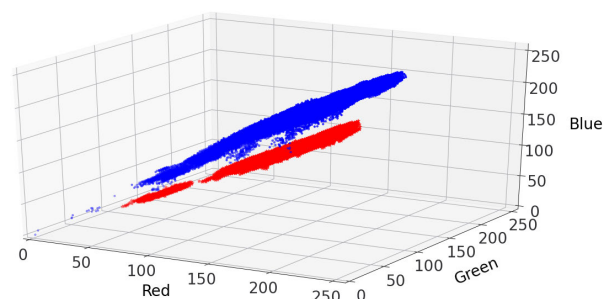
The image processing was carried out using a Python class, which cuts the image based on the relative, configurable coordinates of the vertices of the rectangle of the portion of the image to be preserved. This class is also able to visualize the pixels of the image in the RGB space in order to study which strategies are best for distinguishing the pixels of the bread sheet from those of the background. An example of images resulting from viewing in RGB space is shown in Figure 5, in the case of the camera placed at the exit from the

<sup>3</sup><https://www.elastic.co/elastic-stack/>





**FIGURE 5.** Multiple image pixels acquired by the camera placed at the exit from the leavening cell displayed in RGB space (red dots: bread sheet, blue dots: spiral cooler conveyor grid.)



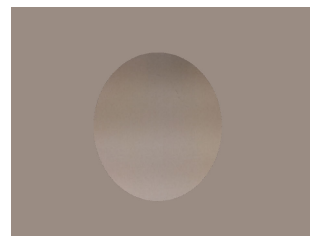
**FIGURE 6.** Multiple image pixels acquired by the camera placed at the entrance from the leavening cell and displayed in RGB space (red dots: bread, blue dots: conveyor belt)

leavening cell, and Figure 6 in the case of the camera placed at the entrance from the leavening cell.

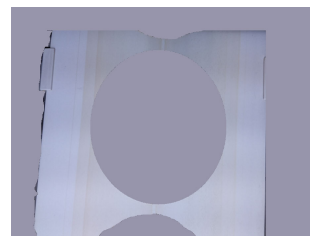
Both at the entrance and at the exit of the leavening room, it can be seen that the points belonging to the bread sheets (colored in red) are linearly separable, except for a few points, from the points belonging to the grid or to the conveyor belt (colored in blue). From Figure 5 and Figure 6 it can be seen how the two sets of points are arranged in parallel and how the difference between the two groups lies in the closest proximity to the blue (higher in the vertical axis) of the pixels not belonging to the sheet. In conclusion, the two graphs show that the two classes of points are almost linearly separable. In case of a linearly separable data set, a SVM with a linear kernel can be an effective solution.

### C. DATASET

Our dataset creation process included: i) color extraction (RGB values) for all pixels; ii) labeling each pixel as either bread or background; iii) removing duplicates – RGB triples with identical values and identical label. Creating the data set was challenging, due to its large size. For this reason we divided it into two subsets. The first one is used as a training set, to fit the classifier model; the second one is used as the, testing set, to evaluate the accuracy of the fitted model by comparing the model prediction with the correct class label.



**FIGURE 7.** Example of image containing only pixels belonging to the dough, used to create the training and testing sets.



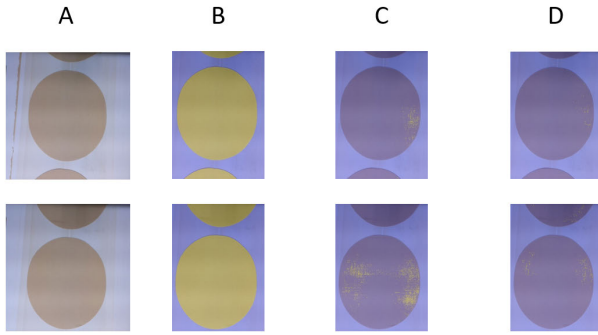
**FIGURE 8.** Example of image containing only pixels belonging to the conveyor belt, used to create the training and testing sets.

In details the data set used to train and to test the algorithm included:

- in the case of images acquired at the entrance from the leavening room:
  - 8,654 training and 1,881,970 testing samples (pixels);
- in the case of images acquired at the exit from the leavening room:
  - 36,577 training and 1,223,743 testing samples.

Two series of data sets have been created, one of which for the camera positioned at the entrance of the second leavening room, and one for the camera positioned at the exit. A set of 10 images was selected for each series, 5 of which are used for training and 5 for testing. In each image, the segmentation of the bread and the background was carried out manually, thus obtaining two images: i) an image containing only pixels belonging to the background (see Fig. 7); ii) an image containing only pixels belonging to the bread disc (see Fig.8).

Each data set consists of two text files, one for training and one for testing. Each instance of the data set corresponds to a pixel of the image represented through 3 features that correspond to the values of red, green and blue channels, according to the representation of the RGB color space; each instance is labelled as 0 (zero) if it does not belong to the dough sheet, and as 1 otherwise. During the creation of the data set, all the duplicates in the training set, i.e., instances that have the same label and the same RGB values, were removed to decrease the size of the training set. As a consequence, the total number of instances (pixels) per image is not constant, but depends on how heterogeneous the pixels are. For example, the data set for the sheet at the entrance contains fewer examples (8,000 against about 37,000 of the sheet at the exit) due to the greater uniformity in the distribution of colors in the case of the incoming sheet. Whereas duplicate removal was applied to training examples,



**FIGURE 9.** Comparison of segmentation results, for two sample images, obtained by different Support Vector Machines trained on different data sets. A) Original image B) Data set created from 5 images with duplicate removal C) Data set created from 10 images with duplicate removal D) Data set created from 5 images (different from the ones of case A) without duplicate removal.

duplicate testing instances were not removed, to account for the actual segmentation accuracy. All of the produced data sets were used to train different SVMs; the results led to the conclusion that further increasing the number of images from which to extract the features does not bring any benefit, and in some cases is counterproductive. In particular, in all cases where duplicates were not removed from the training set, the results were worse than those obtained by removing them.

**D. SUPPORT VECTOR MACHINES**

SVMs are a supervised machine learning approach for classification and regression tasks. For classification problems, starting with a set of  $l$  training examples:

$$\{(\mathbf{x}_i, y_i), i = 1, 2, \dots, l\} \tag{1}$$

where  $\mathbf{x} \in \mathbb{R}^n$  denotes the  $n$ -dimensional feature vector of an instance and  $y$  denotes its class label, a SVM classifier  $f(\mathbf{x})$  implements the following decision function:

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \mathbf{x} + b \tag{2}$$

in which  $\mathbf{w} \in \mathbb{R}^n$  is a vector of feature weights, and  $b$  is the bias term. Eq. (2) can be extended to non-linear classifiers through a non-linear mapping  $\Phi(\cdot)$  to map the instances  $\mathbf{x}$  into a higher dimensional space  $\mathcal{H}$ . The non-linear SVM decision function becomes:

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \Phi(\mathbf{x}) + b \tag{3}$$

For a linearly separable training set,  $\mathbf{w}$  and  $b$  can be found by minimizing the following cost function:

$$\min_{\mathbf{w}} J(\mathbf{w}) = \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} = \frac{1}{2} \|\mathbf{w}\|^2 \tag{4}$$

subject to the following constraints:

$$\begin{aligned} \mathbf{w}^T \cdot \mathbf{w} + b &\geq +1 \text{ for } y_i = +1 \\ \mathbf{w}^T \cdot \mathbf{w} + b &\leq -1 \text{ for } y_i = -1; \quad i = 1, 2, \dots, l \end{aligned} \tag{5}$$

If the training data is not linearly separable, slack variables  $\xi_i$  can be introduced to relax the above constraints, as in the following:

$$\begin{aligned} \mathbf{w}^T \cdot \mathbf{x} + b &\geq +1 - \xi_i \text{ for } y_i = +1 \\ \mathbf{w}^T \cdot \mathbf{x} + b &\leq -1 + \xi_i \text{ for } y_i = -1; \quad i = 1, 2, \dots, l \\ \xi_i &\geq 0; \quad i = 1, 2, \dots, l \end{aligned} \tag{6}$$

In this way the cost function becomes:

$$\min J(\mathbf{w}, \xi) = \frac{1}{2} \|\mathbf{w}\|^2 + C \cdot \sum_{i=1}^l \xi_i \tag{7}$$

where  $C$  is a user-specified parameter that controls the trade-off between margin maximisation (low values of  $C$ ) and training error minimization (high values of  $C$ ). This serves the purpose of avoiding *overfitting*, a situation in which the classifier correctly classifies the training data but it is unable to properly classify data outside the training set. The cost function (7) can be minimized by using the Lagrange multipliers method; it turns out that  $\mathbf{w}$  can be expressed as a linear combination of the  $l$  vectors  $\Phi(x_i)$ :

$$\mathbf{w} = \sum_{i=1}^l \alpha_i \cdot y_i \cdot \Phi(x_i) \tag{8}$$

where  $\alpha_i \geq 0, i = 1, 2, \dots, l$  are the Lagrange multipliers associated with the constraints in (6). By substituting the so defined  $\mathbf{w}$  in the decision function (3) it becomes:

$$f(x) = \sum_{i=1}^l \alpha_i \cdot y_i \cdot \Phi^T(x_i) \cdot \Phi(x) + b \tag{9}$$

To find the values of the Lagrange multipliers  $\alpha_i \geq 0, i = 1, 2, \dots, l$ , the dual form of the the optimization problem (7) is usually considered:

$$\begin{aligned} \max W(\alpha_1, \alpha_2, \dots, \alpha_l) \\ = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \alpha_i \sum_{j=1}^l \alpha_j \cdot \alpha_j \cdot y_i \cdot y_j \cdot \Phi^T(x_i) \cdot \Phi(x_j) \end{aligned} \tag{10}$$

with the following constraints:

$$0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, l \quad \sum_{i=1}^l \alpha_i \cdot y_i = 0 \tag{11}$$

Since the dual cost function  $\mathbf{W}$  is convex and quadratic in terms of the unknown variables  $\alpha_i$ , it can be solved numerically through quadratic programming. It is worth noting that the terms  $\Phi^T(x_i) \cdot \Phi(x_j)$  in the Eq. (10) can be expressed as a function  $K(x_i, x_j)$ , called Kernel function. Note that suitable kernel functions can be used directly, without the need of explicitly defining and computing the non-linear mapping  $\Phi(\mathbf{x})$ . Using this so-called “kernel trick” allows to use non-linear functions as a kernel in a simpler way. Some examples of non-linear kernels for SVMs are:

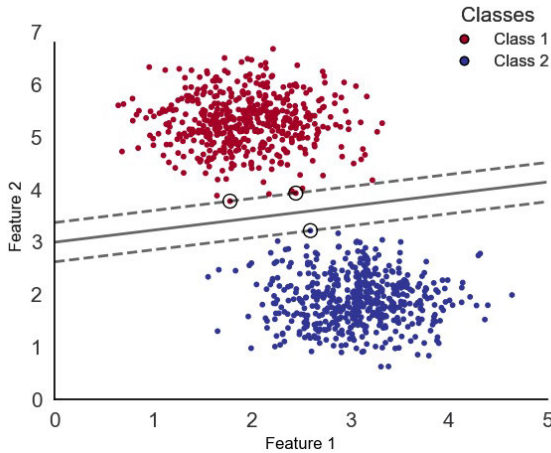


FIGURE 10. Example of a linear support vector machine for a linearly separable training set.

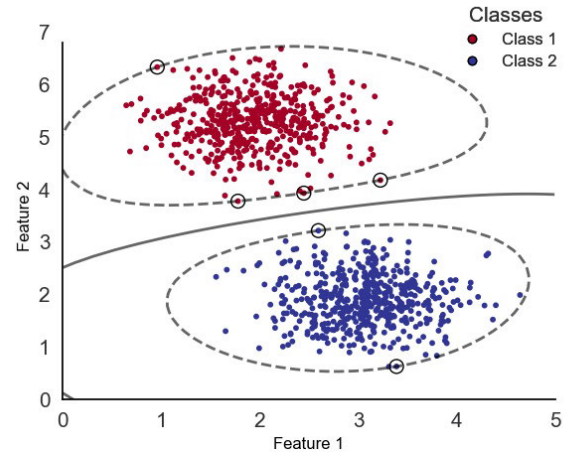


FIGURE 11. Example of non-linear SVM with a Gaussian Kernel, for the same training set of Fig. 10.

- Polynomial kernel of degree  $d$  (see Figure 12):

$$K(x, y) = (x^T \cdot y + C)^d \quad (12)$$

- Gaussian kernel with standard deviation  $\sigma$  (see Figure 11)

$$K(x, y) = \exp\left(-\frac{\|x - z\|^2}{2\sigma^2}\right) \quad (13)$$

In Figure 10 we show an example of a two-class problem (Class 1 in red and Class 2 in blue) with two features and a linearly separable training set. The class boundary (separation hyperplane) found by a SVM with a linear kernel is shown as the continuous line that separates the two groups of points. The two dashed lines represent the margins: as can be seen, they intersect some of the points belonging to the two classes (two points for class 1 and one point for class 2). These points are the support vectors that give the algorithm its name. These points are important because for the purposes of classifier training their positions are the only ones that matter: it turns out that they are the only ones for which the corresponding Lagrange multiplier in Eq. (8) is different from zero. This means that the positions of the points that do not cross the margins do not count for the determination of the separation hyperplane, which depends only on the support vectors.

For our image segmentation problem we evaluated SVMs with different kernels. Generally the best results are obtained with a Gaussian kernel, but the small improvements over linear classifiers do not compensate for the longer computation time. Two linear kernel SVMs were therefore used. Table 3 compares the performance of SVMs with different kernels: Gaussian, linear, linear with correction of the weights of the two classes, polynomials of different degrees, and sigmoidal.

These performances apply to the algorithm that evaluates the outgoing sheets. The performance parameters evaluated are the following:

- Accuracy: number of correct predictions against total predictions;

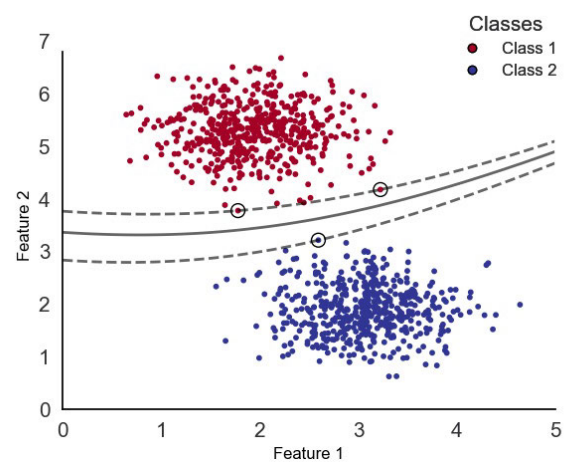



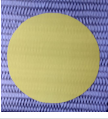
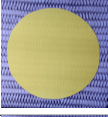
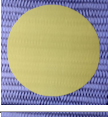
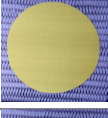
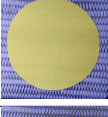
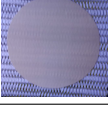
FIGURE 12. Example of SVM non-linear SVM with a third-degree polynomial kernel, for the same training set of Fig. 10.

- Precision: number of times bread is predicted correctly out of the total number of times bread is predicted;
- Recall: number of times bread is predicted correctly out of the real amount of bread elements (pixels);
- F1-score: harmonic mean of precision and recall;
- Cohen’s  $K$ : measure of the degree of agreement: it is excellent if it is greater than 0.8, little or no agreement if it is less than or equal to 0.4. If it is negative it means that the classification is worse than assigning the classes at random;
- Average computation time: average time needed to classify all the pixels of an image.

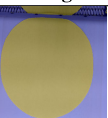
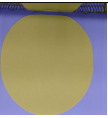
Given the performance obtained in the outgoing case, i.e. the worst case, a comparison was also made for the classification of the incoming sheet. In this case only Gaussian and linear kernels were considered (see Table 4). By virtue of the results obtained it was decided to apply two SVMs with a linear kernel to perform classification.

The SVM proposed in this work aims at classifying each pixel of an input image as either dough or background, using

**TABLE 3.** Comparison of SVM classifiers with different kernels for the outgoing dough.

Type	Image	Accuracy	Precision	Recall	F1 score	Cohen's K	Average computation time [s]
Gaussian		0.9743	0.9199	0.8699	0.8942	0.8796	45.531
Linear		0.9685	0.8680	0.8817	0.8748	0.8568	0.080
Linear W1		0.9702	0.8841	0.8760	0.88	0.863	12.429
Linear W10		0.9544	0.7363	0.9890	0.8441	0.8181	17.8097
Polynomial 2		0.97	0.8766	0.8841	0.8804	0.8632	10.982
Polynomial 6		0.9739	0.9117	0.8756	0.8933	0.878	10.093
Sigmoid		0.7237	0.0133	0.0166	0.0148	-0.1436	316.954

**TABLE 4.** SVM comparison with different kernels for the ingoing dough.

Type	Image	Accuracy	Precision	Recall	F1 score	Cohen's K	Average computation time [s]
Gaussian		0.9965	0.9891	1	0.9945	0.992	7.072
Linear		0.9956	0.9862	1	0.9930	0.9898	0.135

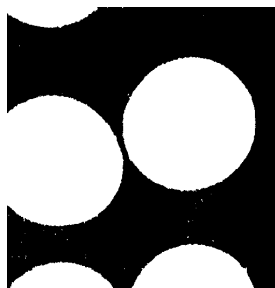
as feature vector the corresponding RGB intensity values, and using a linear kernel that allows to keep processing time well below 1 s per image.

## V. CLASSIFICATION OF THE BREAD SHEETS - OUTPUT SEGMENTATION

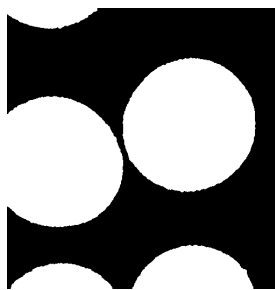
The classification of the bread sheet photo is carried out through the use of a linear SVM trained according to the procedure described in IV-C. Images were acquired and converted into a matrix of dimensions equal to [number of vertical pixels  $\times$  number of horizontal pixels  $\times$  3] containing the BGR values (which are the RGB values with red and blue inverted) used by the OpenCV software library<sup>4</sup> to represent

<sup>4</sup>Open source Python library for computer vision. URL: <https://opencv.org/>

the images. Once the image has been converted into a matrix, the SVM prediction function is applied to the entire matrix, producing a matrix of dimensions [number of vertical pixels  $\times$  number of horizontal pixels] whose elements equal 1, if they are classified as “bread”, or 0, if they are classified as “background”. In this way, each pixel of the analyzed image will be associated with the “bread” or “background” class. The resulting prediction matrix is converted into a BW image as Fig. 13. In Figure 13 some noise can be observed, which is due to false positive classification errors. They can cause problems and prevent the correct operation of the size estimation algorithm. For this reason we use two fundamental morphological operators such as *erosion* and *dilation* in order to eliminate noise or reduce it considerably. *Erosion* consists in reducing the size of the objects present in the image by



**FIGURE 13.** Image obtained by converting the matrix with the per-pixel SVM predictions into a B/W image (black: background, white: dough).



**FIGURE 14.** The image of Fig. 13 after the opening operation with elliptical kernel.

estimating the local minima with respect to the kernel area. The kernel is a matrix composed of ones and zeros arranged to form certain figures (crosses, rectangles, ellipses etc.) [82]. *Dilation*, on the other hand, is an operator that increases the size of the figures contained in the images through the convolution of the image with the kernel [82]. By using erosion in the correct way, the spots due to noise can be eliminated and it is also possible to reduce the irregularity of the edges. However, this also reduces the size of the image for which we want to estimate the parameters. To solve this problem, we just apply a dilation of the same amount immediately after the erosion. This operation, in which an erosion is applied followed by a dilation of the same entity and with the same kernel, is called opening [83]. By applying an opening with an elliptical kernel of  $7 \times 7$  dimensions on the image in Figure 13 we obtained a clean image as in Figure 14. Here the noise has been completely removed and the irregularity of the edges is less marked, even if it did not completely disappear.

#### A. FILTERING

Although in most cases the proposed algorithm is able to correctly segment the dough sheet, in particular cases, such as the presence of waste or the absence of dough, the SVM incorrectly classifies a large fraction of pixels, which results into an excess of points classified either as bread or as sheets. In order to skim the images subject to this issue, the ratio of the pixels classified as dough is computed.

A value of zero means that all the pixels are classified as background, and a value of 1 means that all the pixels are classified as dough. Several images were analyzed to identify the pixel ratio range associated with each category of images.

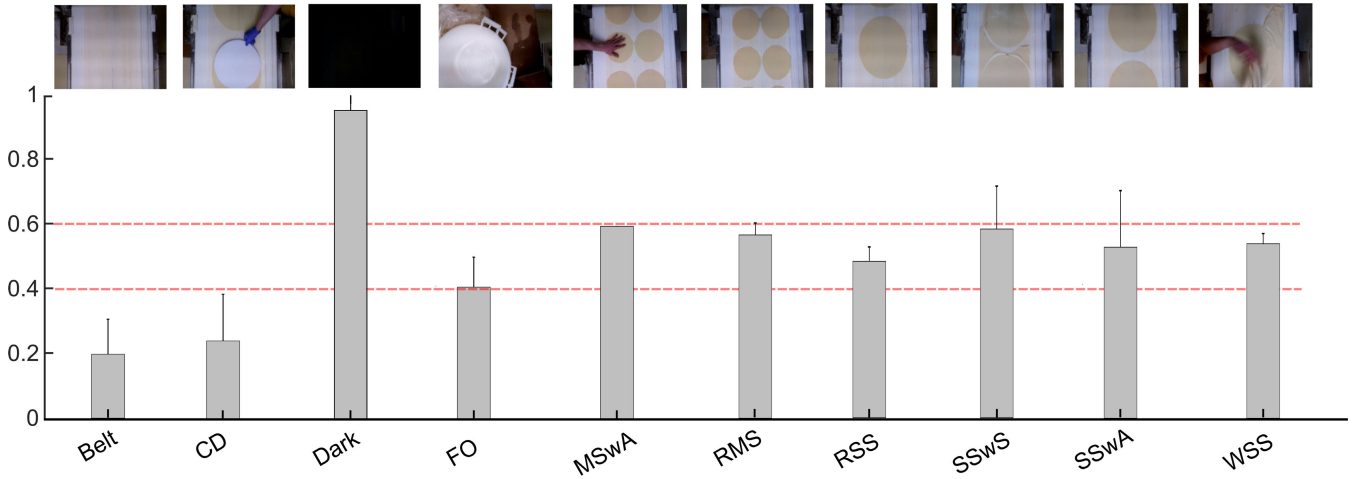
The categories identified were ten and for a better comprehension we show an example image at the top of Figure 15:

- Right single sheet (RSS): the sheet of bread for the 36 cm format photographed in its entirety and classified correctly;
- Right multiple sheet (RMS): bread sheets for the 18 cm format photographed in groups, usually of six sheets, and classified correctly;
- Wrong single sheet (WSS): single sheet for the 36 cm format not classified correctly. Generally in these cases the sheet is not photographed in its entirety;
- Belt: The photo only contains the conveyor belt without bread;
- Single sheet with shrinkage (RSSws): with 36 cm sheets there is some shrinkage that prevents the correct shape of the sheet from being determined;
- Foreign objects (FO): in the photo there are objects foreign to the bread making process, while the sheets are not present;
- Cardboard Disc (CD): operators use a cardboard disc with a diameter of 36 cm as a reference for the size of the sheets. The presence of this disc in the photos prevents the correct classification of the sheet;
- Single sheet with an arm (SSwA): in the photo there is a 36 cm sheet but also an arm that interferes with the classification;
- Multiple sheet with an arm (MSwA): the same problem occurs as in the previous case but for the sheets with a diameter of 18 cm;
- Dark: the photo was taken in the dark and consequently the classification result was inconsistent with the expected result.

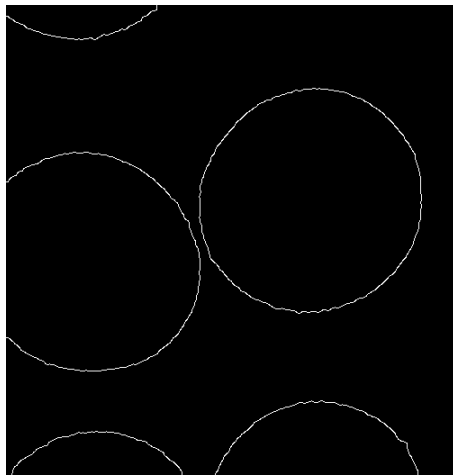
Figure 15 shows how the filter works for each different case. The basic requirement is for all right sheets (RSS, RMS) to pass the filter. This means the superimposed standard deviations of these two cases must be inside the range between the lower limit and the higher limit (the red lines in the figure). The actual filter adds a little extra margin to those limits to avoid eventual right sheets with a pixel ratio too near to the limits, determined from the samples, to be accidentally discarded. Since the filter is designed to avoid discarding the RMS and RSS cases, there is a limit in what it can filter (since cases like FO, MSwA and WSS are not filtered at all), but it's able to filter completely the more extreme cases (Belt, CD, Dark) and almost completely the most frequent anomalies during the bread image capture (SSwS, SSwA).

#### B. SIZE ESTIMATION

Once the image was cleaned, the edges of the bread sheet were estimated with the Canny algorithm, an edge detection algorithm that determines the edges by estimating the intensity gradients of the image in different directions and selecting those that they get the maximum variation [80]. By applying Canny to the image in Figure 14, an image containing only the edges is obtained. Once the image with



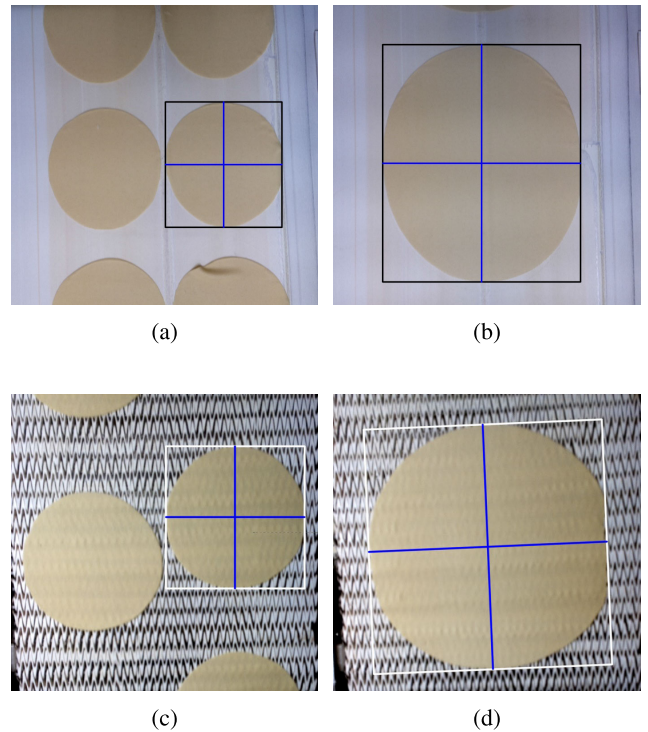
**FIGURE 15.** Overview of pixel-based filter results on different scenarios. The vertical bars represent the pixel ratio, with superimposed standard deviation range for each case. The horizontal dotted red lines are the maximum (0.6) and minimum (0.4) values allowed.



**FIGURE 16.** Edges detected with the Canny algorithm.

the outlines has been obtained, it is necessary to determine the pixels belonging to each of the curves present in the image and select the one of interest which, as in the case of Figure 16, is the only closed curve present. The first operation is carried out through the application of Suzuki’s algorithm [81], used by the computer vision OpenCV library, which is a border tracing algorithm that takes a black and white image as input and outputs a list of arrays containing each the contour points of one of the curves in the image. As for the selection of the curve to be used to evaluate the geometric parameters, we selected the curve with the largest area: in this way a closed curve is always selected on which the geometric parameters can be calculated.

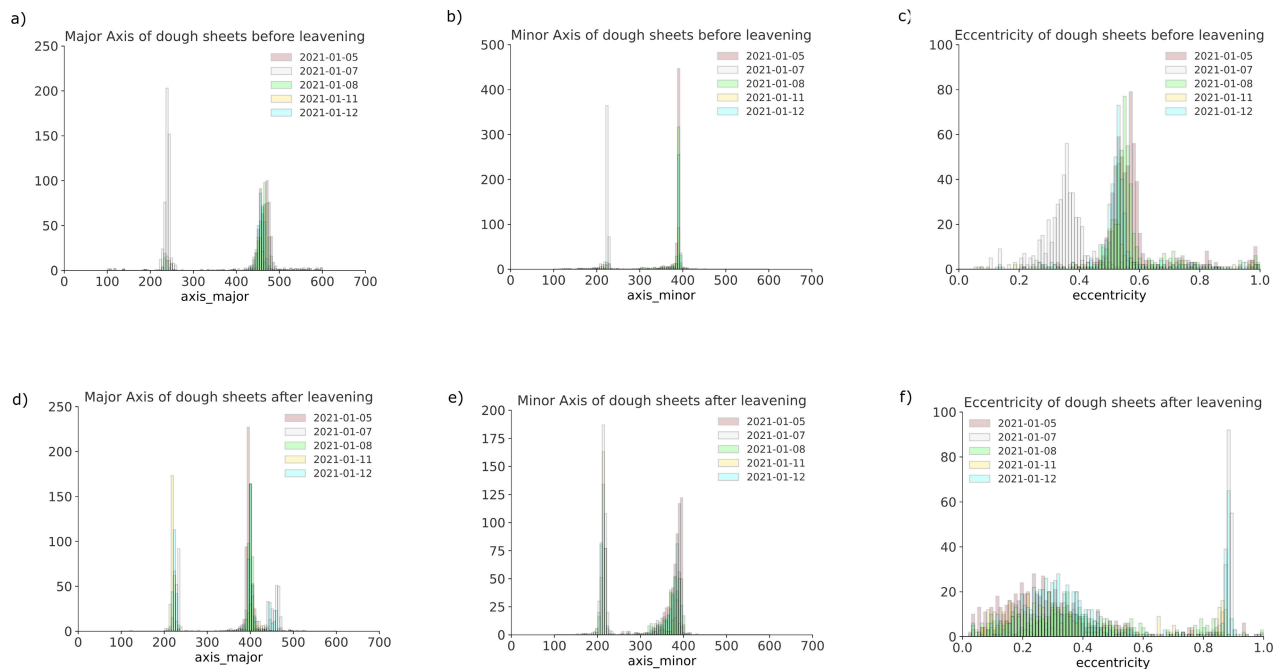
Despite the application of morphological parameters, the shape of the contours was still very irregular. For this reason, instead of the actual contours, their convex envelope was considered from which a curve with a slightly greater area than that of the contours is obtained. The diameter estimation error due to this replacement, however, does not exceed 6 pixels in the worst cases (metal belt) with average diameters



**FIGURE 17.** Major and minor axes extraction results. (a) 180 mm diameter dough before leavening; (b) 360 mm diameter dough before leavening; (c) 180 mm diameter dough after leavening; (d) 360 mm diameter dough after leavening.

of 308 for the 180 mm format sheet and 513 for the 360 mm format sheet. This leads to a maximum error of 2.2% of the pixel size measure in the worst case scenario and less than 1% of the pixel size measure in the best case scenario. In all other scenarios the error has an average value of 1.2%.

To estimate the geometric parameters, the minimum bounding rectangle that encloses the convex envelope of the curve was determined. This was done through the OpenCV library which uses the rotating caliper algorithm to find the target rectangle. This rectangle was represented by five values



**FIGURE 18. a) Major axis before leavening. b) Minor axis before leavening. c) Eccentricity before leavening. d) Major axis after leavening. e) Minor axis after leavening. f) Eccentricity after leavening. All axes dimensions are in mm.**

corresponding to the  $x$  and  $y$  coordinates of the center, the width and height and the angle of inclination of the lower side with respect to the horizontal axis. Once this rectangle was obtained, it was possible to obtain the length of the major and minor axes of the sheet which coincided, respectively, with the width and height of the rectangle.

## VI. EXPERIMENTAL RESULTS

In this section, we demonstrate the segmentation results of the proposed algorithm for segmenting Carasau bread sheets. Fig. 17 shows some results of the extraction of the major and minor axes from the dough sheet images acquired during the Carasau production. The size estimation was carried out on images acquired at two different stages of the industrial process and in two different rooms: the dough production room, where the dough sheets are moving on a white conveyor belt (see Figs. 17a and 17b), and at the exit of the leavening cell, where the leavened sheets are deployed on a metallic belt (see Figs. 17c and 17d). The sizes of the sheets, in both cases, are approximately equal to the sides of the minimum area bounding rectangle computed by the proposed algorithm. The blue axes shown in Fig. 17a-17d have the same size of the sides. From a qualitative analysis, the finding reported in Fig. 17a-17d demonstrate that the proposed color segmentation algorithm can cope with the industrial scenario in hand.

After having demonstrated, from a qualitative point of view, that our computer vision solution can effectively perform as a visual inspection tool, we refined our analysis by performing preliminary quantitative assessment over some days of Carasau production. In detail, we extracted the sizes of the Carasau bread sheets during a normal

production, for five days, both at the dough production and after the leavening, and analyzed the distribution of the geometrical features. In Fig. 18 we report the histogram of major and minor axes, and the eccentricity, derived during the investigated period. From the analysis of Fig. 18, it can be noticed that the minor and major axes tend to concentrate around an average value, and exhibit a Gaussian-like distribution. The peaks in Figs. 18a, 18b, 18d and 18e are reasonable and expected. Indeed, the larger sheet size ( $\sim 36$  cm of nominal diameter), during the production, before baking and cutting (see Fig. 1), is known to have a size larger than 400 mm. Furthermore, our stakeholders reported that values up to 500 mm can be observed. On the other hand, the smaller sheet ( $\sim 18$  cm of nominal diameter) during the production is known to have a characteristic size comprised between 200 and 300 mm. The reasons behind this large variability in the minor and major axes sizes before leavening was due to the fact that, during normal operations, the operators manually adjust production parameters, such as belt velocities, to tune the manufacturing. After the leavening, by observing Figs. 18d and 18e, it can be noticed that the chemo-physical process of fermentation results in a reduction of the major axis, while determining a more random variation of the minor axis dimension. As regards the eccentricity, which is supposed to tend to zero for a perfectly circular Carasau sheet, we can highlight that the large (all days) and small (January 7th, 11th, 12th 2021) sheets enter the leavening room with a very different eccentricity (0.4-0.7 vs. 0.1-0.4, as shown in Fig. 18c), but then tend to present overlapping distributions, as can be seen from Fig. 18f. It is worth noting that the eccentricity peak around 0.9 in Fig. 18f can be ascribed to an anomalous case in the production,

in which two small sheets exited the leavening cell stuck to each other and are erroneously classified as a single object, thus overestimating the actual sizes. In all the other cases, the eccentricity after leavening is lower than that before the fermentation.

These results are promising and allow us to forecast the use of the proposed computer vision algorithm as a tool to empower Carasau bread manufacturing.

## VII. CONCLUSION AND DISCUSSION

Over the years, there has been a significant amount of effort on the segmentation of colour food images. In this paper, we addressed the problem of automatic image segmentation with application to the partial automation of the production process of a traditional Sardinian flat bread in order to assure quality control. In our case study, we are faced, on the one hand, with a relatively simple segmentation problem with an object of almost uniform color over a uniform background (dough sheets entering the leavening room); on the other hand we also have a problem with non-uniform background characterized by a non-regular pattern (dough sheets exiting the leavening room). In this paper, we proposed an approach that explores and combines new segmentation models in order to get final reliable segmentation results, leading to a robust algorithm for segmenting food images from the background. As a first option, our approach segments and classifies the individual pixels into bread sheet or background, based on their color in the RGB space. Since it is not easy to manually define a classification rule, we have opted to use a machine learning-based approach that requires manual segmentation of a certain number of images to build a training set.

Our study focused on one of the most critical activities for obtaining an efficient degree of automation: the estimation of the size and shape of the bread sheets during the production phase, in order to study the shape variations undergone by the sheet depending on some environmental and production variables. We demonstrated that food quality control and assurance in the producing lines may be supported by using a machine learning algorithm, based on Support Vector Machines, for the segmentation and estimation of the shape and size of images of Carasau bread. This knowledge can thus be used to create a system capable of predicting the quality of the shape of the dough produced, and therefore to empower the production process. The proposed algorithm proved to be fast and accurate in estimating the size of the bread sheets in the various scenarios that occurred over a year of acquisitions. The maximum error made by our algorithm equals 2.2% of the pixel size in the worst scenario and 1.2% otherwise.

Possible future directions include improving the method to get a smooth curve from the curve of the contour with a more precise one or investigating more refined ways of finding the measures in millimetres from the pixel ones. In future work, the use of this algorithm, as a part of a prediction system, will be investigated with environmental and production data. It will provide bread operators with

directions about undesired outcomes in the bread production process and allow them to take corrective actions far in advance.

## ACKNOWLEDGMENT

The authors would like to thank Fabrizio Di Napoli e Antonio Loddo from the M.F.M. of Urrai Salvatora & C. S.N.C for participating in this study and for the information they freely provided. (Matteo Baire and Katuscia Mannaro contributed equally to this work.)

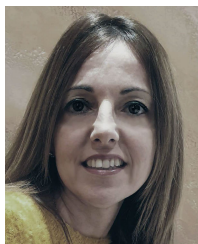
## REFERENCES

- [1] M. B. Lodi, N. Curreli, A. Melis, E. Garau, F. Fanari, A. Fedeli, A. Randazzo, G. Mazzarella, and A. Fanti, "Microwave characterization and modeling of the Carasau bread doughs during leavening," *IEEE Access*, vol. 9, pp. 159833–159847, 2021.
- [2] L. Cocco, K. Mannaro, R. Tonelli, L. Mariani, M. B. Lodi, A. Melis, M. Simone, and A. Fanti, "A blockchain-based traceability system in agri-food SME: Case study of a traditional bakery," *IEEE Access*, vol. 9, pp. 62899–62915, 2021.
- [3] D. S. Nguyen, "Application of Bayesian networks for product quality management in a multistage manufacturing process," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IIEEM)*, Dec. 2015, pp. 1402–1406.
- [4] M. Baire, A. Melis, M. B. Lodi, L. Lodi, A. Fanti, and G. Mazzarella, "Empowering traditional Carasau bread production using wireless sensor network," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2021, pp. 1–4.
- [5] H. D. Cheng, X. H. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Advances and prospects," *Pattern Recognit.*, vol. 34, no. 12, pp. 2259–2281, Dec. 2001.
- [6] R. Manzke, C. Meyer, O. Ecabert, J. Peters, N. J. Noordhoek, A. Thiagalangam, V. Y. Reddy, R. C. Chan, and J. Weese, "Automatic segmentation of rotational X-ray images for anatomic intra-procedural surface generation in atrial fibrillation ablation procedures," *IEEE Trans. Med. Imag.*, vol. 29, no. 2, pp. 260–272, Feb. 2010.
- [7] Y. Li, J. Zhang, P. Gao, L. Jiang, and M. Chen, "Grab cut image segmentation based on image region," in *Proc. IEEE 3rd Int. Conf. Image, Vis. Comput. (ICIVC)*, Jun. 2018, pp. 311–315.
- [8] C. Narmatha and S. Manimurugan, "A new approach for iris image identification using modified contour segmentation," in *Proc. Int. Conf. Green Comput. Commun. Electr. Eng. (ICGCCCE)*, Mar. 2014, pp. 1–7.
- [9] D.-M. Tsai and S.-C. Lai, "Independent component analysis-based background subtraction for indoor surveillance," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 158–167, Jan. 2009.
- [10] P. Arias, A. Pini, G. Sanguinetti, P. Sprechmann, P. Cancela, A. Fernandez, A. Gomez, and G. Randall, "Ultrasound image segmentation with shape priors: Application to automatic cattle rib-eye area estimation," *IEEE Trans. Image Process.*, vol. 16, no. 6, pp. 1637–1645, Jun. 2007.
- [11] J. M. Ponce, A. Aquino, B. Millan, and J. M. Andújar, "Automatic counting and individual size and mass estimation of olive-fruits through computer vision techniques," *IEEE Access*, vol. 7, pp. 59451–59465, 2019.
- [12] M. Baire, A. Melis, M. B. Lodi, P. Tuveri, C. Dachena, M. Simone, A. Fanti, G. Fumera, T. Pisanu, and G. Mazzarella, "A wireless sensors network for monitoring the Carasau bread manufacturing process," *Electronics*, vol. 8, no. 12, p. 1541, Dec. 2019.
- [13] A. Gongal, M. Karkee, and S. Amatya, "Apple fruit size estimation using a 3D machine vision system," *Inf. Process. Agricult.*, vol. 5, no. 4, pp. 498–503, 2018.
- [14] N. Senthilkumaran and S. Vaithegi, "Image segmentation by using thresholding techniques for medical images," *Comput. Sci. Eng., Int. J.*, vol. 6, no. 1, pp. 1–13, Feb. 2016.
- [15] D. Liu and J. Yu, "Otsu method and K-means," in *Proc. 9th Int. Conf. Hybrid Intell. Syst.*, vol. 1, Sep. 2009, pp. 344–349.
- [16] Y.-J. Zhang, "An overview of image and video segmentation in the last 40 years," in *Advances in Image and Video Segmentation*. Hershey, PA, USA: IGI Global, 2006, pp. 1–16.
- [17] J. Almotiri, K. Elleithy, and A. Elleithy, "A multi-anatomical retinal structure segmentation system for automatic eye screening using morphological adaptive fuzzy thresholding," *IEEE J. Transl. Eng. Health Med.*, vol. 6, pp. 1–23, 2018.



- [18] H. Liang, H. Jia, J. Ma, X. Peng, and Z. Xing, "Modified grasshopper algorithm-based multilevel thresholding for color image segmentation," *IEEE Access*, vol. 7, pp. 11258–11295, 2019.
- [19] J. Liao, Y. Wang, D. Zhu, Y. Zou, S. Zhang, and H. Zhou, "Automatic segmentation of crop/background based on luminance partition correction and adaptive threshold," *IEEE Access*, vol. 8, pp. 202611–202622, 2020.
- [20] D. A. Zebari, D. Q. Zeebaree, A. M. Abdulazeez, H. Haron, and H. N. A. Hamed, "Improved threshold based and trainable fully automated segmentation for breast cancer boundary and pectoral muscle in mammogram images," *IEEE Access*, vol. 8, pp. 203097–203116, 2020.
- [21] L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surv.*, vol. 24, no. 4, pp. 325–376, Dec. 1992.
- [22] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. Hoboken, NJ, USA: Wiley, 2009.
- [23] A. Bouti, M. A. Mahraz, J. Riffi, and H. Tairi, "Robust system for road sign detection and recognition using template matching," in *Proc. Intell. Syst. Comput. Vis. (ISCV)*, Apr. 2017, pp. 1–4.
- [24] A. Poonsri, N. Aimjirakul, T. Charoenpong, and C. Sukjamsri, "Teeth segmentation from dental X-ray image by template matching," in *Proc. 9th Biomed. Eng. Int. Conf. (BMEiCON)*, Dec. 2016, pp. 1–4.
- [25] X. Zhou, Y. Wang, and Q. Zhu, "A fast and robust visual detection method with binary template matching for glass bottle body," in *Proc. Chin. Control Conf. (CCC)*, Jul. 2019, pp. 7672–7676.
- [26] P. Swaroop and N. Sharma, "An overview of various template matching methodologies in image processing," *Int. J. Comput. Appl.*, vol. 153, no. 10, pp. 8–14, Nov. 2016.
- [27] Z. Zlatev, M. Baeva, and P. Nikolova, "Segmentation of microscopic images of bacteria in Bulgarian yoghurt by template matching," *Sci. J. Fac. Technol. Yambol*, vol. 4, no. 3, pp. 211–225, 2016.
- [28] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.
- [29] Y.-J. Zhang, "Half century for image segmentation," in *Encyclopedia of Information Science and Technology*, 3rd ed. Pennsylvania, PA, USA: IGI Global, 2015, pp. 5906–5915.
- [30] A. Mehnert and P. Jackway, "An improved seeded region growing algorithm," *Pattern Recognit. Lett.*, vol. 18, no. 10, pp. 1065–1071, 1997. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167865597001311>
- [31] J. Fan, G. Zeng, M. Body, and M.-S. Hacid, "Seeded region growing: An extensive and comparative study," *Pattern Recognit. Lett.*, vol. 26, no. 8, pp. 1139–1156, Jun. 2005.
- [32] V. R. P. Borges, M. C. F. D. Oliveira, T. G. Silva, A. A. H. Vieira, and B. Hamann, "Region growing for segmenting green microalgae images," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 15, no. 1, pp. 257–270, Jan. 2018.
- [33] F. Y. Shih and S. Cheng, "Automatic seeded region growing for color image segmentation," *Image Vis. Comput.*, vol. 23, no. 10, pp. 877–886, Sep. 2005.
- [34] L. S. Davis, "A survey of edge detection techniques," *Comput. Graph. Image Process.*, vol. 4, no. 3, pp. 248–270, 1975.
- [35] R. Muthukrishnan and M. Radha, "Edge detection techniques for image segmentation," *Int. J. Comput. Sci. Inf. Technol.*, vol. 3, no. 6, p. 259, 2011.
- [36] Y. Liu, Z. Xie, and H. Liu, "An adaptive and robust edge detection method based on edge proportion statistics," *IEEE Trans. Image Process.*, vol. 29, pp. 5206–5215, 2020.
- [37] S. S. Al-Amri, N. V. Kalyankar, and S. D. Khamitkar, "Image segmentation by using edge detection," *Int. J. Comput. Sci. Eng.*, vol. 2, no. 3, pp. 804–807, 2010.
- [38] G. Shrivakshan and C. Chandrasekar, "A comparison of various edge detection techniques used in image processing," *Int. J. Comput. Sci. Issues*, vol. 9, no. 5, p. 269, Sep. 2012.
- [39] R. Maini and H. Aggarwal, "Study and comparison of various image edge detection techniques," *Int. J. Image Process.*, vol. 3, no. 1, pp. 1–11, Feb. 2009.
- [40] M. Sharifi, M. Fathy, and M. T. Mahmoudi, "A classified and comparative study of edge detection algorithms," in *Proc. Int. Conf. Inf. Technol., Coding Comput.*, Apr. 2002, pp. 117–120.
- [41] H. Kagaya, K. Aizawa, and M. Ogawa, "Food detection and recognition using convolutional neural network," in *Proc. 22nd ACM Int. Conf. Multimedia*. New York, NY, USA: ACM, Nov. 2014, pp. 1085–1088, doi: 10.1145/2647868.2654970.
- [42] B. Debska and B. Guzowska-Swider, "Application of artificial neural network in food classification," *Analytica Chim. Acta*, vol. 705, nos. 1–2, pp. 283–291, 2011. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0003267011008622>
- [43] F.-P. An and Z.-W. Liu, "Medical image segmentation algorithm based on feedback mechanism CNN," *Contrast Media Mol. Imag.*, vol. 2019, pp. 1–13, Aug. 2019.
- [44] Z. Xu, S. Zuo, E. Y. Lam, B. Lee, and N. Chen, "AutoSegNet: An automated neural network for image segmentation," *IEEE Access*, vol. 8, pp. 92452–92461, 2020.
- [45] D. Karimi and S. E. Salcudean, "Reducing the Hausdorff distance in medical image segmentation with convolutional neural networks," *IEEE Trans. Med. Imag.*, vol. 39, no. 2, pp. 499–513, Feb. 2020.
- [46] Y. Jiang, L. Duan, J. Cheng, Z. Gu, H. Xia, H. Fu, C. Li, and J. Liu, "JointRCNN: A region-based convolutional neural network for optic disc and cup segmentation," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 2, pp. 335–343, Feb. 2020.
- [47] G. Guan, S. Wang, and A. Wee-Chung Liew, "Lip image segmentation based on a fuzzy convolutional neural network," *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 7, pp. 1242–1251, Jul. 2020.
- [48] G. Ö. Yigit and B. M. Özyildirim, "Comparison of convolutional neural network models for food image classification," *J. Inf. Telecommun.*, vol. 2, no. 3, pp. 347–357, Jul. 2018.
- [49] A. Kamilaris and F. X. Prenafeta-Boldú, "A review of the use of convolutional neural networks in agriculture," *J. Agricult. Sci.*, vol. 156, no. 3, pp. 312–322, 2018.
- [50] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping Multidimensional Data*. Berlin, Germany: Springer, 2006, pp. 25–71.
- [51] O. A. Abbas, "Comparisons between data clustering algorithms," *Int. Arab J. Inf. Technol.*, vol. 5, no. 3, pp. 320–325, 2008.
- [52] A. Alush and J. Goldberger, "Hierarchical image segmentation using correlation clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 6, pp. 1358–1367, Jun. 2016.
- [53] X. Jia, T. Lei, P. Liu, D. Xue, H. Meng, and A. K. Nandi, "Fast and automatic image segmentation using superpixel-based graph clustering," *IEEE Access*, vol. 8, pp. 211526–211539, 2020.
- [54] H. Huang, F. Meng, S. Zhou, F. Jiang, and G. Manogaran, "Brain image segmentation based on FCM clustering algorithm and rough set," *IEEE Access*, vol. 7, pp. 12386–12396, 2019.
- [55] E. Ungor and A. Ozmen, "Coarse segmentation with GDD clustering using color and spatial data," *IEEE Access*, vol. 8, pp. 144880–144891, 2020.
- [56] L. Zhou and W. Wei, "DIC: Deep image clustering for unsupervised image segmentation," *IEEE Access*, vol. 8, pp. 34481–34491, 2020.
- [57] M.-N. Wu, C.-C. Lin, and C.-C. Chang, "Brain tumor detection using color-based K-means clustering segmentation," in *Proc. 3rd Int. Conf. Intell. Inf. Hiding Multimedia Signal Process. (IHH-MSP)*, Nov. 2007, pp. 245–250.
- [58] M. Mignotte, "Segmentation by fusion of histogram-based K-means clusters in different color spaces," *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 780–787, May 2008.
- [59] X. Bai, Y. Zhang, H. Liu, and Y. Wang, "Intuitionistic center-free FCM clustering for MR brain image segmentation," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 5, pp. 2039–2051, Sep. 2019.
- [60] S. Banerjee, A. Choudhary, and S. Pal, "Empirical evaluation of K-means, bisecting K-means, fuzzy C-means and genetic K-means clustering algorithms," in *Proc. IEEE Int. WIE Conf. Electr. Comput. Eng. (WIECON-ECE)*, Dec. 2015, pp. 168–172.
- [61] S. R. Dubey, P. Dixit, N. Singh, and J. P. Gupta, "Infected fruit part detection using K-means clustering segmentation technique," *Int. J. Interact. Multimedia Artif. Intell.*, vol. 2, no. 2, p. 65, 2013.
- [62] X. Zheng, Q. Lei, R. Yao, Y. Gong, and Q. Yin, "Image segmentation based on adaptive K-means algorithm," *EURASIP J. Image Video Process.*, vol. 2018, no. 1, pp. 1–10, Dec. 2018.
- [63] J. Siswanto, A. S. Prabuwo, A. Abdullah, and B. Idrus, "Automatic image segmentation using Sobel operator and k-means clustering: A case study in volume measurement system for food products," in *Proc. Int. Conf. Sci. Inf. Technol. (ICSITech)*, Oct. 2015, pp. 13–18.
- [64] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, Jun. 1991.
- [65] J. B. T. M. Roerdink and A. Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," *Fundam. Inform.*, vol. 41, nos. 1–2, pp. 187–228, 2000.
- [66] A. Kornilov and I. Safonov, "An overview of watershed algorithm implementations in open source libraries," *J. Imag.*, vol. 4, no. 10, p. 123, Oct. 2018. [Online]. Available: <https://www.mdpi.com/2313-433X/4/10/123>

- [67] H. P. Ng, S. H. Ong, K. W. C. Foong, P. S. Goh, and W. L. Nowinski, "Medical image segmentation using K-means clustering and improved watershed algorithm," in *Proc. IEEE Southwest Symp. Image Anal. Interpretation*, Mar. 2006, pp. 61–65.
- [68] V. Grau, A. U. J. Mewes, M. Alcaniz, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Trans. Med. Imag.*, vol. 23, no. 4, pp. 447–458, Apr. 2004.
- [69] Y. Yuan, Z. Zhu, H. Yu, and W. Zhang, "Watershed-based superpixels with global and local boundary marching," *IEEE Trans. Image Process.*, vol. 29, pp. 7375–7388, 2020.
- [70] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Jul. 1995.
- [71] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intell. Syst. Appl.*, vol. 13, no. 4, pp. 18–28, Jul. 1998.
- [72] M. Gong, Y. Qian, and L. Cheng, "Integrated foreground segmentation and boundary matting for live videos," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1356–1370, Apr. 2015.
- [73] Y. Wang, Y. Sun, P. Lv, and H. Wang, "Detection of line weld defects based on multiple thresholds and support vector machine," *NDT E Int.*, vol. 41, no. 7, pp. 517–524, Oct. 2008.
- [74] L. Wang, P. C. Pedersen, E. Agu, D. M. Strong, and B. Tulu, "Area determination of diabetic foot ulcer images using a cascaded two-stage SVM-based classification," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 9, pp. 2098–2109, Sep. 2017.
- [75] X.-Y. Wang, X.-J. Zhang, H.-Y. Yang, and J. Bu, "A pixel-based color image segmentation using support vector machine and fuzzy C-means," *Neural Netw.*, vol. 33, pp. 148–159, Sep. 2012.
- [76] A. J. M. Timmermans and A. A. Hulzebosch, "Computer vision system for on-line sorting of pot plants using an artificial neural network classifier," *Comput. Electron. Agricult.*, vol. 15, no. 1, pp. 41–55, 1996. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0168169995000569>
- [77] E. S. Gadelmawla, "Computer vision algorithms for measurement and inspection of spur gears," *Measurement*, vol. 44, no. 9, pp. 1669–1678, Nov. 2011.
- [78] C. Ighathinathane, L. O. Pordesimo, and W. D. Batchelor, "Major orthogonal dimensions measurement of food grains by machine vision using ImageJ," *Food Res. Int.*, vol. 42, no. 1, pp. 76–84, Jan. 2009.
- [79] Q. Zeng, C. Liu, Y. Miao, S. Fei, and S. Wang, "A machine vision system for continuous field measurement of grape fruit diameter," in *Proc. 2nd Int. Symp. Intell. Inf. Technol. Appl.*, Dec. 2008, pp. 1064–1068.
- [80] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.
- [81] S. Suzuki and K. Be, "Topological structural analysis of digitized binary images by border following," *Comput. Vis., Graph., Image Process.*, vol. 30, no. 1, pp. 32–46, 1985. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0734189X85900167>
- [82] J. Hanjun and S. Huali, "Rendering fake soft shadows based on the erosion and dilation," in *Proc. 2nd Int. Conf. Comput. Eng. Technol.*, vol. 6, 2010, pp. V6-234–V6-236.
- [83] M. Guan, H. Ren, and Y. Ma, "Multi-scale morphological filtering method for preserving the details of images," in *Proc. Asia-Pacific Conf. Comput. Intell. Ind. Appl. (PACIIA)*, vol. 2, Nov. 2009, pp. 417–420.



**KATIUSCIA MANNARO** received the Engineering degree (*summa cum laude*) from the University of Cagliari, Italy, in 2001, and the Ph.D. degree in electronic engineering and computer science with a thesis on "Adopting Agile Methodologies in Distributed Software Development," in 2008. She worked as a Postdoctoral Fellow with the Department of Electrical and Electronic Engineering (DIEE), University of Cagliari, from 2010 to 2017, and since 2019, she has been holding a postdoctoral fellowship position with the Department of Mathematics and Computer Science.

Her research interests include blockchain technologies, software modeling, agile and lean methodologies, smart cities. She has teaching experience as a Lecturer in the field of the software engineering and her research is published in a large number of conference proceedings, articles in books, and international journals.



**MATTEO BAIRE** received the Laurea degree in electrical engineering from the University of Cagliari, in 2010. He worked as a Research Assistant with the Electromagnetic Group, University of Cagliari, from 2017 to 2020, where he is currently a Technologist. His research interests include artificial intelligence, computer vision, wireless sensor networks, food engineering, and sensor processing.



**ALESSANDRO FANTI** (Member, IEEE) received the Laurea degree in electronic engineering and the Ph.D. degree in electronic engineering and computer science from the University of Cagliari, Cagliari, Italy, in 2006 and 2012, respectively. He worked as a Postdoctoral Fellow with the Electromagnetic Group, University of Cagliari, from 2013 to 2016, where he is currently an Assistant Professor. He has coauthored more than 100 scientific contributions published in

international journals, conference proceedings, and book chapters. His research interests include the use of numerical techniques for modes computation of guiding structures, optimization techniques, analysis and design of waveguide slot arrays, analysis and design of patch antennas, radio propagation in urban environment, modeling of bio-electromagnetic phenomena, and microwave exposure systems for biotechnology and bio-agriculture. He is a member of the IEEE Antennas and Propagation Society, the Italian Society of Electromagnetism, and the Interuniversity Center for the Interaction between Electromagnetic Fields and Biosystems. From 2020 to 2023, he acts as a Principal Investigator of the IAPC Project, funded with five million euros by the Italian Ministry of Economic Development (MISE), within the AGRIFOOD PON I&C (2014–2020). He is also an Associate Editor of the IEEE JOURNAL OF ELECTROMAGNETICS, RF AND MICROWAVES IN MEDICINE AND BIOLOGY (J-ERM).



**MATTEO BRUNO LODI** (Graduate Student Member, IEEE) received the bachelor's degree in biomedical engineering from the University of Cagliari, Cagliari, in 2016, and the master's degree in biomedical engineering from the Politecnico di Torino, Turin, Italy, in 2018. He is currently pursuing the Ph.D. degree in electronic engineering and computer science with the University of Cagliari.

His research interests include the modeling of bioelectromagnetic phenomena, especially hyperthermia treatment, the study, manufacturing, and synthesis of magnetic biomaterials for tissue engineering applications, and the use of microwave for biotechnology and environmental applications. He was awarded as a Young Scientists at General Assembly and Scientific Symposium of URSI 2020 and 2021. He has been appointed as a Representative for the Young Professionals of IEEE Region 8 Nanotechnology Council. He is a member of the Editorial Board of the IEEE Future Directions Technology Policy and Ethics newsletter.



**LUCA DIDACI** (Member, IEEE) received the Ph.D. degree in electronic engineering and computer science from the University of Cagliari, Italy, in 2005. Since 2006, he has been an Assistant Professor in computer engineering with the University of Cagliari. His research interests include methodologies and applications of statistical pattern recognition, and include multiple classifier systems, adversarial machine learning, and analysis of EEG signals using machine

learning techniques. On these topics, he has coauthored more than 40 papers in international journals and conference proceedings. He is a member of the Pattern Recognition and Applications Research Group and also the Italian chapter of the International Association for Pattern Recognition (IAPR). He served as the Guest Editor for the Special Issue on Biometric Presentation Attacks: Handcrafted Features Versus Deep Learning Approaches—*Pattern Recognition Letters* (Elsevier).



**ANDREA RANDAZZO** (Senior Member, IEEE) received the Laurea degree in telecommunication engineering and the Ph.D. degree in information and communication technologies from the University of Genoa, Genoa, Italy, in 2001 and 2006, respectively. He is currently a Full Professor in electromagnetic fields with the Department of Electrical, Electronic, Telecommunication Engineering, and Naval Architecture, University of Genoa. He has coauthored the book *Microwave*

*Imaging Methods and Applications* (Artech House, 2018) and more than 270 articles published in journals and conference proceedings. His research interests include the field of microwave imaging, inverse-scattering techniques, numerical methods for electromagnetic scattering and propagation, electrical tomography, and smart antennas.



**GIUSEPPE MAZZARELLA** (Senior Member, IEEE) received the degree (*summa cum laude*) in electronic engineering from the Università Federico II of Naples, in 1984, and the Ph.D. degree in electronic engineering and computer science, in 1989. In 1990, he became an Assistant Professor with the Dipartimento di Ingegneria Elettronica, Università Federico II of Naples. Since 1992, he has been with the Dipartimento di Ingegneria Elettrica ed Elettronica, Università di Cagliari, first

as an Associate Professor and then, since 2000, as a Full Professor, teaching courses in electromagnetics, microwave, antennas and remote sensing. He is the author or coauthor of over 90 articles in international journals. He is a reviewer for many EM journals. His research interests include efficient design of large arrays of slots, power synthesis of array factor, with emphasis on inclusion of constraints, microwave holography techniques for the diagnosis of large reflector antennas, use of evolutionary programming for the solution of inverse problems, and in particular problems of synthesis of antennas and periodic structures.



**GIORGIO FUMERA** (Member, IEEE) received the Ph.D. degree in electronic engineering and computer science from the University of Cagliari, Italy, in 2002. Since 2010, he has been an Associate Professor in computer engineering with the University of Cagliari. His research interests include methodologies and applications of statistical pattern recognition, and include multiple classifier systems, adversarial machine learning, and computer vision for intelligent video surveillance.

On these topics, he has coauthored more than 100 papers in international journals and conference proceedings. He is a member of the Pattern Recognition and Applications Research Group. He is an Associate Editor of the *Pattern Recognition* and the *Pattern Analysis and Applications* journals. He is a member of the IEEE Computer Society and the Italian chapter of the International Association for Pattern Recognition (IAPR).

...



**ALESSANDRO FEDELI** (Member, IEEE) received the B.Sc. and M.Sc. degrees in electronic engineering and the Ph.D. degree in science and technology for electronic and telecommunications engineering from the University of Genoa, Genoa, Italy, in 2011, 2013, and 2017, respectively. He is currently an Assistant Professor with the Department of Electrical, Electronic, Telecommunications Engineering, and Naval Architecture, University of Genoa. His research activities, carried out with the Applied Electromagnetics Laboratory, are mainly focused on the development and the application of computational methods for the solution of forward and inverse scattering problems, and electromagnetic imaging. He has coauthored more than 130 scientific contributions published in international journals and conference proceedings. He is a member of the IEEE Antennas and Propagation Society, the Italian Society of Electromagnetism, and the Interuniversity Center for the Interaction between Electromagnetic Fields and Biosystems.



**LUISANNA COCCO** received the Ph.D. degree in electronic and computer engineering from the University of Cagliari, Italy, in 2013. Since June 2013, she collaborates with the Agile Group, which is a research group in the field of software engineering with the Università degli Studi di Cagliari. Her research interests include the modeling of the complex systems, in particular economic and financial systems with heterogeneous agents. Since 2014, she has been extending her research

interests to the modeling and simulation of the cryptocurrencies systems publishing papers of great significance for the scientific community and in general to the blockchain technology.