

Article

# Deep Cellular Automata-Based Feature Extraction for Classification of the Breast Cancer Image

Surasak Tangsakul \*  and Sartra Wongthanavasul \* 

College of Computing, Khon Kaen University, Khon Kaen 40002, Thailand

\* Correspondence: surasak.tang@kkumail.com (S.T.); wongsar@kku.ac.th (S.W.)

**Abstract:** Feature extraction is an important step in classification. It directly results in an improvement of classification performance. Recent successes of convolutional neural networks (CNN) have revolutionized image classification in computer vision. The outstanding convolution layer of CNN performs feature extraction to obtain promising features from images. However, it faces the overfitting problem and computational complexity due to the complicated structure of the convolution layer and deep computation. Therefore, this research problem is challenging. This paper proposes a novel deep feature extraction method based on a cellular automata (CA) model for image classification. It is established on the basis of a deep learning approach and multilayer CA with two main processes. Firstly, in the feature extraction process, multilayer CA with rules are built as the deep feature extraction model based on CA theory. The model aims at extracting multilayer features, called feature matrices, from images. Then, these feature matrices are used to generate score matrices for the deep feature model trained by the CA rules. Secondly, in the decision process, the score matrices are flattened and fed into the fully connected layer of an artificial neural network (ANN) for classification. For performance evaluation, the proposed method is empirically tested on BreakHis, a popular public breast cancer image dataset used in several promising and popular studies, in comparison with the state-of-the-art methods. The experimental results show that the proposed method achieves the better results up to 7.95% improvement on average when compared with the state-of-the-art methods.

**Keywords:** breast cancer; feature extraction; deep cellular automata; image classification



**Citation:** Tangsakul, S.;

Wongthanavasul, S. Deep Cellular Automata-Based Feature Extraction for Classification of the Breast Cancer Image. *Appl. Sci.* **2023**, *13*, 6081. <https://doi.org/10.3390/app13106081>

Academic Editor: Yu-Dong Zhang

Received: 9 April 2023

Revised: 9 May 2023

Accepted: 12 May 2023

Published: 15 May 2023

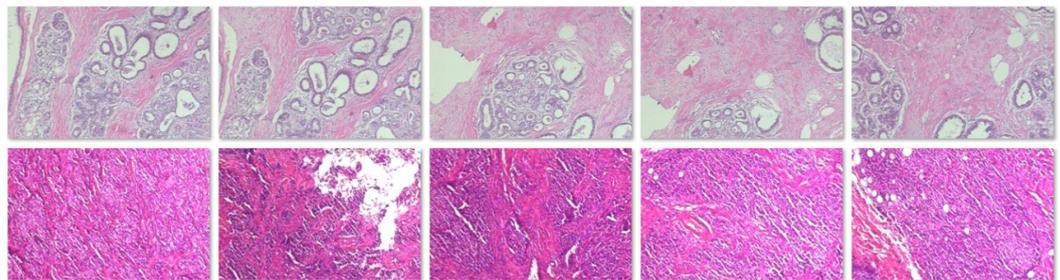


**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The natural performance mechanism of the human body sustains a cycle of revival processes and balances the growth and mortality rates of its cells. However, the abnormal growth of a few cells can cause cancer, which grows and distributes across the human body. According to the [1] report, several types of cancer develop in the human body. For women, breast cancer is one of the leading causes of death. Various factors cause breast cancer: human anatomy (man or woman), family history, age, obesity, and food [2]. However, abnormal cell growth can be classified into two significant case types: benign and malignant cases. Benign cases are considered to be noncancerous or nonlife-threatening, but they occasionally transform into cancer. The human immunity system is known as a “sac” [1]; it is a usual mechanism that can separate benign tumors from normal cells and then remove them from the body. For malignant cases, cancer originates from abnormal cell growth that might suddenly distribute or attack neighboring cells. If cells exhibit abnormal growth, the nuclei of the tissue might rapidly become more prominent than normal tissue, which might lead to a frightening case in the future. To identify breast tumor cases, immediately imaging the target area can help a doctor perform further diagnoses. Two groups of medical photography techniques are based on skin perforation and tissue condition: (i) noninvasive methods, i.e., computer-aided tomography (CAT), ultrasound, X-ray, and magnetic resonance imaging (MRI); and (ii) invasive methods, i.e., histopathological imaging (aka. biopsy imaging).

A histopathological image is a microscopic tissue image that is widely used for disease analysis and predominates as the gold standard for cancer diagnosis [3,4]. Such images give valuable and essential information that doctors can intensively investigate to find the current circumstances of the patient. Until recently, histopathology images were difficult to discover and retrieve and were unavailable to the scientific community. Therefore, most histopathology image analyses were performed on small datasets, especially for breast cancer images. To attenuate this gap, ref. [5] proposed a dataset called BreakHis that has become one of the largest and most popular datasets, with 7909 breast histopathology images from 82 patients divided into two main classes: benign and malignant. Figure 1 shows examples of breast histopathology images obtained from this dataset.



**Figure 1.** Example breast cancer histopathology images derived from the BreakHis dataset.

Feature extraction is an important process used to identify relevant information for any classifier. There are several approaches for feature extraction; Pixel-level features that determine the features from the pixel in the image, Global features that are used to describe the entire image, and Local features that are used only to describe the exciting part of the image. The first-level classification performance of six handcrafted feature vectors was also provided as baselines by [5] to distinguish between benign and malignant cases. These handcrafted features are based on the pixel-level features achieved by state-of-the-art descriptors, including grey-level co-occurrence matrices (GLCMs) [6], local binary patterns (LBPs) [7], threshold adjacency statistics (PFTASs) [8], local phase quantization (LPQ) [9], completed LBPs (CLBPs) [10], and oriented FAST and rotated BRIEF (ORB) [11].

In machine learning (ML), it is one of the most popular approaches to getting computers to act without being explicitly programmed. In many domains, such as natural language processing (NLP) [12–14], computer vision (CV) [15–17], and deep learning (DL) [18–22], numerous problems have been solved with increasing accuracy and dramatic results. Several feature extraction methods are adopted in medical imaging, especially for histopathological image classification. For example, the effectiveness of an ensemble classification method by adopting a set of support vector machines (SVMs) and random subspace ensembles of multilayer perceptrons (MLP) was explored by [23]. In this work, the set of SVM classifiers relied on two learning features: the curvelet transformation and the LBP. Then, the final classifier was built by merging the SVM and artificial neural network (ANN) classifiers. Six different feature descriptors were evaluated by [5] with four classifiers, i.e., 1-nearest neighbor (1-NN), SVM, quadratic linear analysis (QDA), and random forest (RF) classifiers. Additionally, an ensemble method to determine the local characteristics of breast cancer tumors was proposed by [24]. The model was executed by aggregating a vector of locally aggregated descriptors (VLAD) on the Grassmann vector. The VLAD was represented as a collection of multidimensional and spatially changing signals on a higher-order linear dynamic analysis system. Furthermore, a classification method that relied on an unsupervised learning approach was proposed by [25] to address the different qualities of histology images, and by learning a domain-invariant space, it is attempted to extract the feature vectors of benign images. In these works, the BreakHis dataset was also used to evaluate the performance of the corresponding methods.

CA and digital image join the common architecture. Hence, image processing tasks to extract features are common to CA processing provided the appropriate rules. In addition,

CA is a universal computation, equivalent to Turing machines, meaning that CA can compute any computing tasks given the appropriate rules. On this basis, we propose CA's based model for feature extraction in breast cancer images for classification as opposed to the outstanding CNN deep learning in the state-of-the-art compared methods. Feature extraction conducted in the convolution layer in CNN for image classification is a promising technique in literature. However, it faces drawbacks in computational complexity and overfitting problems. This research aims at solving these drawbacks in feature extraction by using a CA-based model, which is totally different from the approach in CNN. In breast cancer image classification, we use the magnification of histopathological images of breast cancer as usual practices in cancer diagnosis.

This paper proposes a deep cellular automata (CA)-based feature extraction for breast cancer image classification by focusing on histopathological images. The main contributions of this paper are summarized as follows.

(1) We propose a novel feature extraction method for image classification tasks by using the deep CA approach.

(2) We present a customized case in which the proposed model can deal with breast cancer image classification.

The remainder of the paper is organized as follows. In the next section, we review the related works on the breast cancer image classification task. Section 3 presents the proposed framework for the deep CA and the classification process. In Section 4, the details of the experiments and their results are comprehensively presented. Finally, Section 5 provides the conclusion of the paper and suggestions for future works.

## 2. Background

### 2.1. Deep Feature Learning in Images

In recent years, deep learning approaches have been widely used in several applications, especially for classification problems. A convolutional neural network (CNN) is one of the most popular deep learning models that is typically composed of two main parts; feature learning (i.e., convolution and pooling layer) and classification (i.e., fully connected layer) [18–21,26–29]. Several papers on breast cancer detection and diagnosis have been published; for example, the stacked sparse autoencoder (SSAE) algorithm was proposed by [30] to analyze breast cancer histopathology images. In the training process, they adopted a greedy strategy to optimize the SSAE, trained each hidden layer, and then fed the output to the next layer. In addition, a model that pretrained on ImageNet and built from several images for mitosis classification was proposed by [31]. In this work, three fully connected layer configurations were also presented to separately improve their robustness. Finally, multiple probabilities were obtained by the training process and averaged as the final output. Another model, a deep learning-based framework to discover and locate unusual areas in a mammogram was proposed by [28]. A segmentation algorithm that combines different types of probabilistic models was proposed by [32]. The algorithm adopts a structured SVM that considers prior locations, a deep belief network, and a Gaussian mixture model for mammographic image segmentation. Furthermore, ref. [29] explored a classification model utilizing DeCaf features (also known as neural codes) for breast histopathological images by applying AlexNet [19] to build a deep learning network called CaffeNet. The outputs from the network layers of CaffeNet were used as DeCaf features for further classification. Besides, a deep learning approach based on crowd annotations (called AGGNet) was presented by [26], this model combines a multiscale convolutional neural network (CNN) architecture with an aggregation layer (AL) to directly handle data aggregation. To cope with insufficient of training images [27] developed a transfer learning approach for a deep CNN (DCNN) model. This learning model was trained for the mass segmentation of mammographic images. Moreover, ref. [19] presented a pretrained AlexNet network that was tuned by a fine-tuning strategy that used images from the ImageNet database [33] for training, and a deep learning model relying on a convolutional architecture called a convolutional sparse autoencoder (CSAE) was proposed by [34]. The model aims to

solve the pixelwise labeling problem and to extract more abstract features from unlabeled data derived from mammogram images. To learn the latent bilateral features from the 3D volume data of digital breast tomosynthesis (DBT) [35] presented a self-taught deep learning model by using a 3-D multiview DCNN. The volumes of interest (VOIs) of the source and target are first transformed by the geometric VOI transformation function before feeding them to a DCNN model. Then, high-level latent bilateral features are obtained by those DCNN structures. Another DCNN was proposed by [36] to classify three classes of mammogram images (normal, benign, and malignant). First, the region of interest (ROI) is determined to enhance the image contrast. Then, the CNN is trained on three image groups: the original ROI images, the images reproduced by a nonnegative matrix factorization (NMF) function, and the images reproduced by a statistical self-similarity function. These functions are used to cope with overfitting and to achieve increased classification accuracy. The concept of a multiple-instance learning (MIL) framework was proposed by [37] to investigate the affinity of MIL, which relies on the analysis of histopathology images for a breast cancer patient diagnosis case without the need for labelling all instances. In addition, ref. [38] proposed a learning method in combination with a deep transfer network and deep convolutional generative adversarial network (DCGAN) to cope with imbalanced breast cancer image dataset. In this approach, image augmentation is performed by the DCGAN, which is adopted only on a small class dataset to increase the number of images in the dataset. Then, the balanced dataset is fed to a VGG16 deep transfer network pretrained on ImageNet. Besides, ref. [39] suggested the hybrid ML model to address the issue of class imbalance. The model uses the kernelized weighted extreme learning machine and pre-trained ResNet50 to assist in the histopathology-based computer-aided diagnosis of breast cancer. The histopathological image was divided into non-overlapping patches, and then ResNet50 identified features for each patch and fed them to the kernelized weighted extreme learning machine (KWELM) for classification. Moreover, to better provide and utilize feature information in the histological images, ref. [40] proposed an IDSNet that combines the DenseNet with the squeeze-and-excitation (SENet) module. In breast cancer biopsy specimens containing fine-grain features, the model and classification subnetwork was utilized to extract more channel features and improve the usage of more significant local information. For the classification of histological images of breast cancer that depend on magnification, ref. [41] suggested the pre-trained Xception model. The Xception model utilizes SVM in conjunction with a number of kernels to produce consistent results for all magnification settings as opposed to the handcrafted method. In addition, ref. [42] suggested a method for classifying breast cancer histopathology images using deep semantic and gray-level co-occurrence matrix (GLCM) features. DenseNet201 was used as the fundamental model, which was pre-trained. The deep semantic features, which were subsequently coupled with the three-channel GLCM features, were derived from the final dense block feature of the convolutional layer. Then, the classification process was carried out using the support vector machine (SVM). Besides, the ResNet18 architecture was employed to extract deep features presented by [43]. Several meta-heuristic algorithms, and conventional machine learning algorithms were used to analyze how optimized deep features affect categorization.

## 2.2. Cellular Automata (CA)

CA were first proposed by [44,45] and are mathematical models used to simulate a complex system in terms of a dynamical model in which discrete time and space consist of a regular lattice of cells in any dimension. The operations of CA usually depend on time; the state of a cell depends on the imposed evolution rule relative to the present state and the neighborhood configuration. The CA structure is defined as a set of cells (also known as a neighborhood) relative to the central cell. Several types of CA are available depending on their dimensions and neighborhoods; the basic class contains elementary CA (ECA) or the one-dimensional CA proposed by [46]. However, the most generally used CA are the two-dimensional CA involving Moore and Von Neumann neighborhoods. In this paper, we

simulate the proposed model only on a Moore neighborhood that aims to directly connect to the 8-connected pixels of a digital image structure [47]. We adopt this configuration not only to enhance the image contrast but also to extract the features of breast cancer images. More specifically, the neighborhood  $N(x, y)$  in the range  $r = 1$  can be represented as follows:

$$N(x_0, y_0)^E = \{(x_{-1}, y_0), (x_0, y_0), (x_1, y_0)\} \quad (1)$$

where  $N$  represents a neighborhood function and  $E$  denotes the ECA.

$$N(x_0, y_0)^V = \{(x_0, y_{-1}), (x_{-1}, y_0), (x_0, y_0), (x_1, y_0), (x_0, y_1)\} \quad (2)$$

where  $V$  represents a Von Neumann neighborhood,  $x$  and  $y$  denote the coordinates of the neighborhood, and  $x_0$  and  $y_0$  denote the center points of the neighborhood.

$$N(x_0, y_0)^M = \{(x, y) : |x - x_0| \leq r, |y - y_0| \leq r\} \quad (3)$$

where  $M$  represents the Moore neighborhood,  $x$  and  $y$  denote the position of the neighbor in the neighborhood,  $x_0$  and  $y_0$  are the center points, and  $r$  denotes the range of the neighborhood.

CA models are the dynamical models that have been successfully used in several tasks, especially in image processing scenarios, such as edge detection [48–51], noise filtering [47,52–54], saliency detection [16,17,55], and image segmentation [15,56,57]. For instance, an edge detection algorithm was proposed by [48], who ran a CA model on grayscale images by using a uniform CA rule in a Von Neumann neighborhood to effectively detect image edges. The spatiotemporal CA-based filtering approach (st-CAF) was proposed by [54] to cope with the problem of denoising image sequences; a spatiotemporal neighborhood was adopted to identify a type of noise and tune the algorithm. The saliency detection method proposed by [17] combines global and local information by using multilayer CA. In this model, the skip links and edge penalty terms are added to an encoder-decoder network to transfer information from high-level layers to lower-level layers; the network relies on a CNN structure to determine the global saliency map and generate the foreground and background codebooks. Then, the local saliency map is obtained by utilizing these codebooks. Finally, a multilayer CA framework is applied to produce the final saliency map from the global and local saliency maps.

Among the learning methods that rely on CA approaches, CA learning and prediction (CALP) is an ensemble learning model proposed by [58] to address the handwritten pattern problem in classification tasks. First, the handwritten patterns evolve from various parameters controlled by the rules of CA. Then, various evolutions of the handwritten pattern are used to build classifiers in an aggregate fashion. Furthermore, ref. [21] proposed a CA-based reservoir system based on a deep learning approach to address reservoir computing. The model has two main processes, encoding and decoding, which rely on CA theory. The ECA (or 1-D neighborhood) was adopted as a reservoir to correctly learn the binary pattern from the input layer and map it to the binary output via a readout layer. In addition, in our previous work, ref. [22] proposed a CA-based learning framework called DeepCA that combines a deep learning approach and CA theory. In this work, we implement the framework with the single-image dehazing algorithm to address the prediction problem regarding the proper global light source, as this is a significant part of dehazing algorithms. Furthermore, DeepCA is also used to determine the haze density class, reserved haze parameter, and global atmospheric light ratio; to improve the transmission map and generate the most suitable haze-free image.

### 3. Proposed Method

This section elaborates on the details of the proposed method, including the basics, architecture, and training process of deep CA feature extraction.

#### 3.1. Basics of DeepCA Feature Extraction

In this section, the basics of the proposed DeepCA feature extraction are provided as follows.

**Definition 1.** (*n-layer DeepCA feature extraction*). An *n-layer deep CA feature extraction* is defined as an 8-tuple as follows:

$$\text{DeepCA} = \langle Z^d, Q, N, f^n, R^n, F^n, S, C_{i,n}^t \rangle \tag{4}$$

where  $Z^d$  denotes a coordinate system with  $d$  dimensions;  $Q$  denotes a finite set of states;  $N$  denotes a finite subset of  $Z^d$  called a neighborhood vector;  $f^n$  denotes a local transition function at layer  $n^{\text{th}}$ ;  $R^n = \{r_1, r_2, \dots, r_m\}$  is a rule vector in the totalistic rule space at layer  $n^{\text{th}}$ ;  $F^n$  denotes the feature matrices of layer  $n^{\text{th}}$  generated by the current states of the input matrices, the local transition function ( $f$ ) and the neighborhood ( $N$ );  $S$  denotes the score matrices generated from the feature matrices and the scoring function; and  $C_{i,n}^t$  denotes the  $i^{\text{th}}$  cell at layer  $n^{\text{th}}$  at time  $t$ .

**Definition 2.** (*Evolution rule*). This rule is significant for the operation of CA in determining whether to evolve the current state of a cell to the next generation. Thus, we define a transition function to evolve cells as follows:

$$C_{i,n}^{t+1} = f(C_{i,n}^t, N_{i,n}) \tag{5}$$

where  $C_{i,n}^t$  represents the  $i^{\text{th}}$  cell at the  $n^{\text{th}}$  layer in the current state (or at time  $t$ ),  $f$  denotes the transition function,  $C_{i,n}^{t+1}$  represents the  $i^{\text{th}}$  cell at the  $n^{\text{th}}$  layer in the next state (or at time  $t + 1$ ), and  $N_{i,n}$  denotes the neighborhood configuration of the cell.

For instance, if we have a Moore neighborhood with the range  $r = 1$ , the results of  $C_{i,n}^{t+1}$  in the  $i^{\text{th}}$  cell at the  $n^{\text{th}}$  layer obtained by Equation (5) can be produced as follows:

$$\begin{aligned} C_{i,n}^{t+1} = & \frac{1}{9}(a_0C_{(x,y)}^t + a_1C_{(x+1,y)}^t + a_2C_{(x+1,y+1)}^t \\ & + a_3C_{(x,y+1)}^t + a_4C_{(x-1,y+1)}^t \\ & + a_5C_{(x-1,y)}^t + a_6C_{(x-1,y-1)}^t \\ & + a_7C_{(x,y-1)}^t + a_8C_{(x+1,y-1)}^t) \end{aligned} \tag{6}$$

where  $a_0, a_1, \dots, a_8$  represent the values of the pixels in the image that correspond to the neighborhood.

**Definition 3.** (*Feature matrices*). The multilayer features in DeepCA are defined as feature matrices obtained by the input image and the CA rule vector via the convolution function. In addition, the sizes of the feature matrices also enhance the classification efficiency. Therefore, the feature matrices of the  $n^{\text{th}}$  layer can be obtained by the convolution function as follows:

$$\begin{aligned} F^n &= f_{\text{conv}}(I^n, R^n, s), \\ &= \{f_{r_1}, f_{r_2}, \dots, f_{r_m}\} \end{aligned} \tag{7}$$

where  $f_{\text{conv}}$  is the convolution function,  $I^n$  is an input image at the  $n^{\text{th}}$  layer,  $R^n$  denotes the rule vector at the  $n^{\text{th}}$  layer,  $s$  denotes the number of strides, and  $f_{r_1}, f_{r_2}, \dots, f_{r_m}$  represent the feature matrix obtained by rules  $r_1$  to  $r_m$ .

**Definition 4.** (*Score matrix*). The goal of the training process is to build a reference model consisting of a memory component. In this regard, we define a particular matrix as a memory component called a score matrix ( $S$ ). This matrix can be obtained through any scoring function, e.g., *maxPool* or *softMax*, enabling us to build matrices at the original size or the modified size of the feature matrices as:

$$S = f_{\text{pool}}(F^n, N_{i,n}, s) \tag{8}$$

where  $f_{\text{pool}}$  represents the scoring function,  $F^n$  denotes the feature matrices, and  $N_{i,n}$  denotes a neighborhood of the  $i^{\text{th}}$  cell at the  $n^{\text{th}}$  layer.

**Definition 5.** (DeepCA decision rule). To adopt DeepCA in decision tasks, we consider the difference between the score matrices of an image ( $S_{r_i}$ ) and its class ( $I_{class}$ ). In this regard, the class of an input image is represented as the score matrix of the model ( $S_{model}$ ), which can be determined from a minimum error that can be calculated by a decision function as follows:

$$S_{model} = \min(f_{err}(S_{r_i}, I_{class})) \tag{9}$$

where  $f_{err}$  denotes an error estimation function, such as a mean squared error (MSE) function.

**Definition 6.** (Rule vector). For feature extraction task, the rule vector is the most significant aspect for all layers of DeepCA in terms of tuning the score matrix values. However, several types of CA rules with different rule space sizes are available [59]. In this work, we address the general and totalistic rule spaces; for the general rule space, the rule members consist of entire rules according to their neighborhood configurations. For example, the general rule space of a Moore neighborhood can be formed from rule 0 to  $2^{2^9}$  (or  $1.34 \times 10^{154}$ ) [59]. We formalize all these rules as the rule vector  $R$  in the  $n^{th}$  layer as follows:

$$R^n = \{r_1, r_2, \dots, r_m\} \tag{10}$$

where  $r_i | i = 1$  to  $m$  denote the rule members in the rule vector.  $m$  denotes the size of the rule space of the CA. However, this paper uses  $m = 2^9$  (or 512) based on the totalistic rule type.

Since the general rule space of a Moore neighborhood has a massive size, reducing the rule space is necessary to reduce the time consumption required during the training process and to enable the determination of the rule vector. In order to reduce the number of general rules in the space, we employ the totalistic rule type proposed by [59,60] that limits the rules from  $2^{2^9}$  to only  $2^9$  (or 512) rules [61] (see Figures 2 and 3). Figure 4a shows the particular setting of the totalistic rule type for a Moore neighborhood with approximately  $2^n$  neighbor according to [59–62].

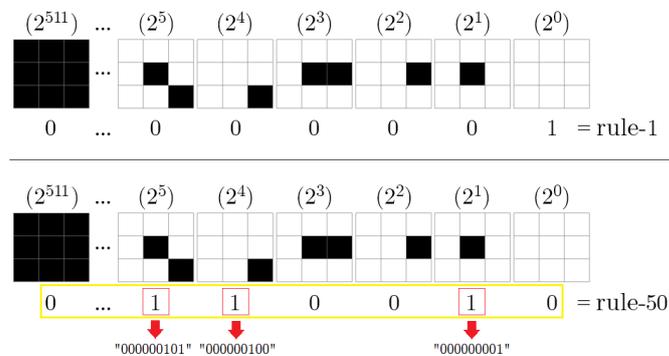


Figure 2. Examples of general rules.

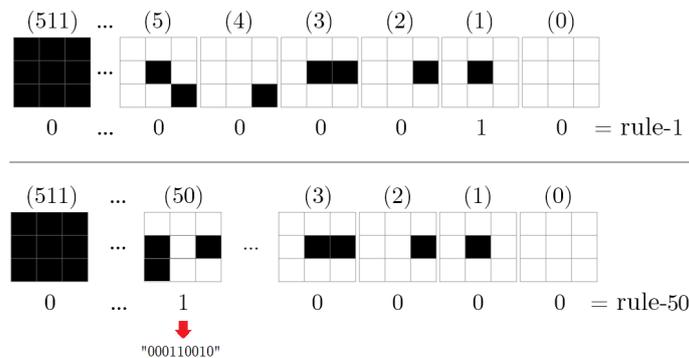
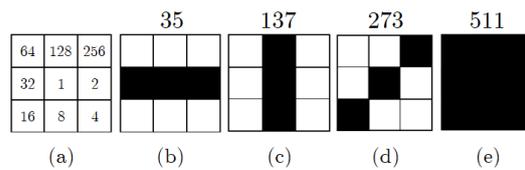


Figure 3. Examples of totalistic rules.



**Figure 4.** Totalistic rules: (a) the neighboring rules’ numbering conventions; (b–e) examples of rules 35, 137, 273, and 511, respectively.

**Definition 7.** (Rule type equivalence). The neighborhood type and the number of possible states are relative to the size of the rule space in general. Considering a Moore neighborhood of size  $3 \times 3$  with two possible states, the range of the general rule space is between 0 and  $2^{512} - 1$  (see Figure 2 (Top)), whereas the range of the totalistic rule space is between 0 (with neighborhood code 000000000) and  $2^9 - 1$  (rule 511 with neighborhood code “111111111”) (see Figure 3 (Top)). In addition, the totalistic rule space is also a part of the general rule space. Therefore, these spaces also have a significant relationship. For instance, to find the neighborhood codes of any rule based on the numbering of rule conventions with neighbors that shown in Figure 4a, the evolution result obtained for input data with rule 50 (or rule code “0 ... 110010”) (see the yellow box in Figure 2)) in the general rule space depends on three neighborhood codes: “000000001”, “000000100”, and “000000101” (see Figure 2 (Bottom)), whereas the evolution result obtained under the totalistic rule depends on only one neighborhood code “000110010” (see Figure 3 (Bottom)). In this regard, the following formalization demonstrates how the general rule and totalistic rule spaces are equivalent.

$$r_{totalistic}(x) \equiv r_{general}(2^{(x)}) \tag{11}$$

For instance, rule 50 in the totalistic rule type (represented as  $r_{totalistic}(50) = “000110010”$ ) is equivalent to rule  $r_{general}(2^{50}) = “000110010”$  in the general type. On the other hand, the rule  $r_{general}(2^1 + 2^4 + 2^5)$  consists of neighborhood codes “000000001”, “000000100”, and “000000101”, which are equivalent to  $r_{totalistic}(1)$ ,  $r_{totalistic}(4)$ , and  $r_{totalistic}(5)$ , respectively.

**Definition 8.** (Rule-0). Even though rule – 0 denotes no operation, the rule – 0 s of the general rule and totalistic rule spaces still exhibit some differences. In this regard, we define  $r_{totalistic}(0) = “000000000”$  as the neighborhood code for matrix convolution and  $r_{general}(0) = null$  as the code for no operation.

**Definition 9.** (Architecture of DeepCA). The proposed framework relies on multilayer CA; therefore, the main architecture of the framework can be represented as:

$$F(x) = f_L(f_{L-1}(f_{L-2}(\dots(f_1(x)))))) \tag{12}$$

where  $F(x)$  denotes the function of the DeepCA architecture with an input  $x$ , and  $f_L$  denotes the functional layer within the  $L^{th}$  layer, which consists of an input layer ( $f_{in}$ ), a convolution layer ( $f_{conv}$ ), a pooling layer ( $f_{pool}$ ), and an output layer ( $f_{out}$ ).

### 3.2. Proposed Framework

This paper proposes a deep CA-based feature extraction approach for image classification. Figure 5 presents an overview of the proposed framework, and it is divided into two stages: training and testing. The images in different magnification factors are performed separately. For the training stage, the images in the training set are resized from  $700 \times 460$  pixels to  $350 \times 320$  pixels. Then, each image is randomly extracted into 100 patches with sizes of  $64 \times 64$  pixels, and the images’ contrast is improved using the GLCM-CA algorithm suggested by [15]. We then feed all image patches to separately train the DeepCA model for each class via the DeepCA training algorithm according to Algorithm 1 and Figure 6. This process determines the best rule vector for the CA and the score matrices. Then, the rule vector and the score matrices are adopted to

reproduce the training set and feed the data to train fully connected neural networks (FCNNs). Finally, during the testing stage, the images in the testing set are subjected to the same process except for the configuration of DeepCA and the FCNNs that apply the rule vector, score matrices and weights obtained by the training process to determine the classification results. The main architecture of DeepCA is shown in Figure 7, which is constructed with multiple layers from  $Layer - 1$  to  $Layer - n$  by applying Equation (12) as  $F(x) = f_{out}(f_{pool}(f_{pool}(f_{pool}(f_{conv}(f_{pool}(f_{conv}(f_{pool}(f_{conv}(f_{in}(x))))))))))$ ). In this architecture, the first functional layer ( $f_{in}$ ) separates each input image into R, G, and B channels and then feeds the image to the next layer. Then, we apply the convolution function ( $f_{conv}$ ) to convolve each image channel with the rule vector in the second layer. The initial parameters of the model in this layer are set as follows: the size of the kernel is set to  $3 \times 3$  according to the Moore neighborhood, the zero padding level is set to 2, and the number of strides is set to 1. Then, the data features extracted by the rules produce the results to be used as feature matrices. Finally, the scoring matrices can be built from the feature matrices obtained from several layers according to Equation (8) and Figure 7; the  $maxPool$  function is employed for all pooling functions ( $f_{pool}$ ), and then the score matrices can be directly obtained by the functional output layer ( $f_{out}$ ).

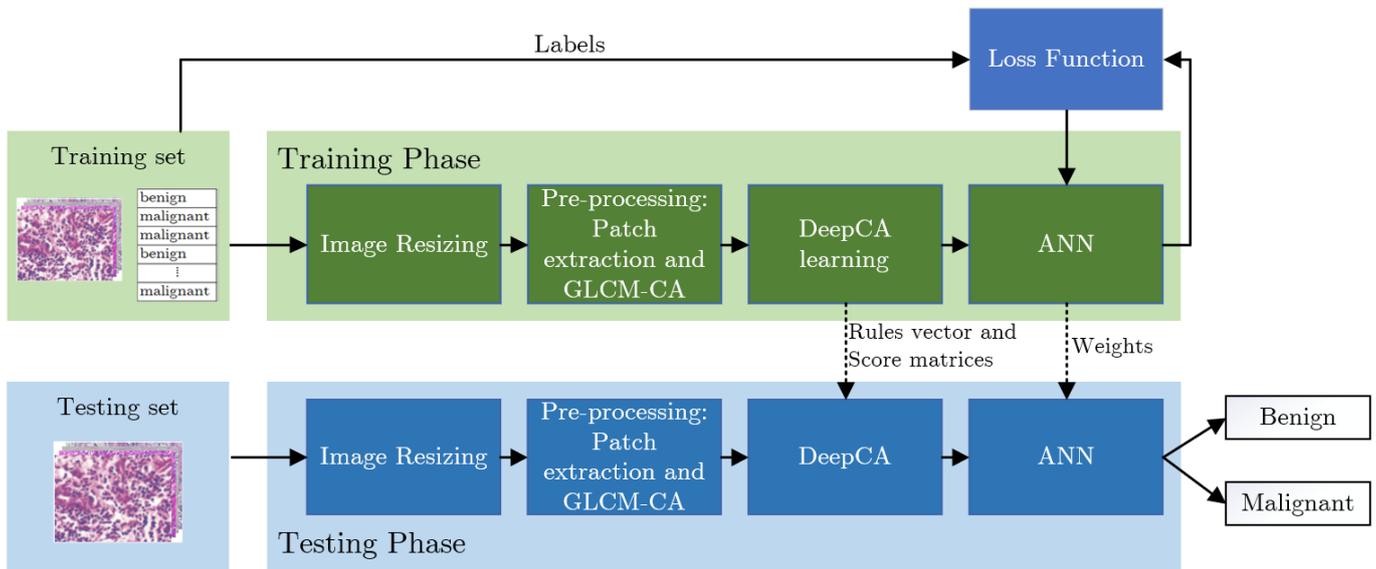


Figure 5. Overview of the proposed framework.

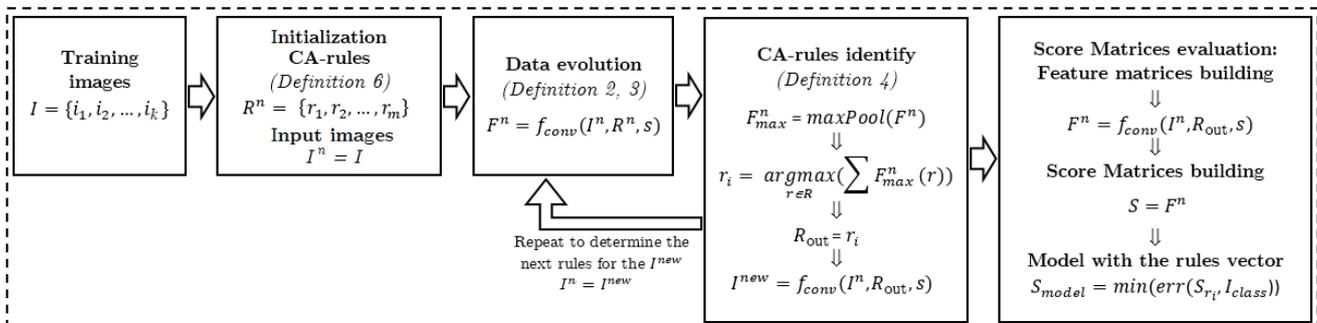
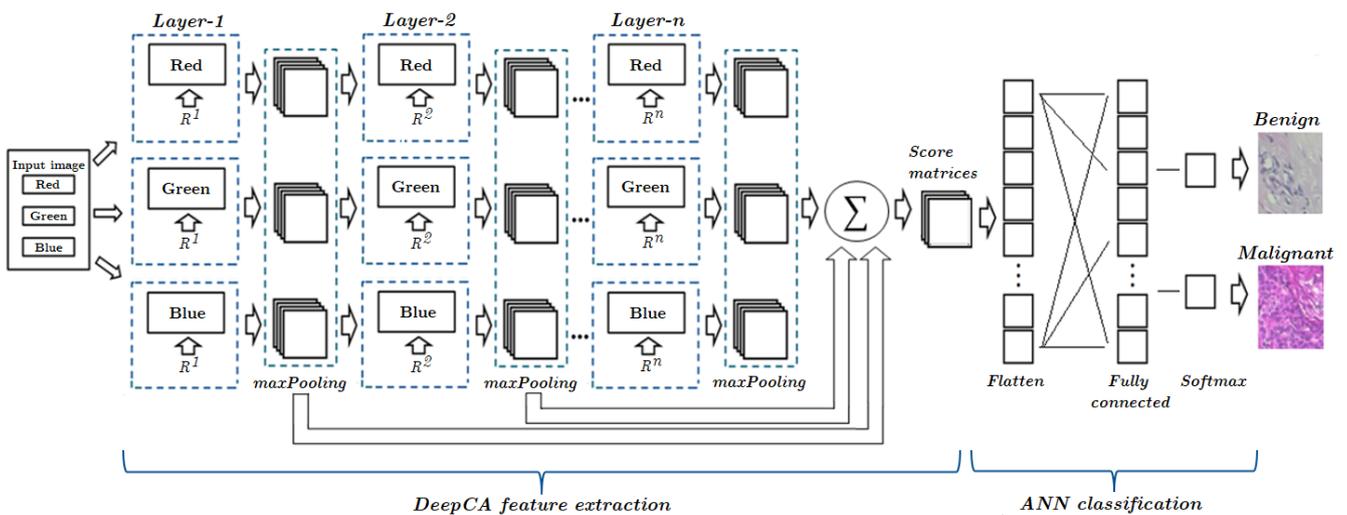


Figure 6. DeepCA training process.

**Algorithm 1:** Training algorithm for DeepCA.

**Input :** Images  $I$  in RGB format.  
**Output:**  $S_{model}$  and  $R_{out}$ .

- 1 Initialization : 2–3
- 2  $N \leftarrow 3 \times 3$
- 3  $I \leftarrow \{i_1, i_2, \dots, i_k\}$
- 4 LOOP Process: 5–26
- 5 **for each layer**  $n$  ( $n = 1, 2, \dots, L$ ) **do**
- 6      $I^n \leftarrow I$  ▷ Assign  $I$  to  $I$  at  $n^{th}$  layer
- 7      $R^n \leftarrow \{r_1, r_2, \dots, r_m\}$  ▷ Initial  $R^n$  by Equation (10)
- 8      $F_{max}^n \leftarrow 0, i \leftarrow 0$  ▷ Initial  $F_{max}^n$  and  $I$  with 0
- 9     **while !convergence do**
- 10         **for each image**  $\in I^n$  **do**
- 11             **forall**  $r \in R^n$  **do**
- 12                  $F^n \leftarrow f_{conv}(I^n, R^n, s)$  ▷ Compute  $F^n$  by Equation (7)
- 13                  $F_{max}^{temp} \leftarrow f_{pool}(F^n, N, s)$
- 14             **end**
- 15              $F_{max}^n \leftarrow sum(F_{max}^n, F_{max}^{temp})$  ▷ Sum  $F_{max}^n$  of all images.
- 16         **end**
- 17          $r_i \leftarrow \arg \max_{r \in R^n}(\sum F_{max}^n(r))$
- 18          $R_{out} \leftarrow r_i$
- 19          $I^{new} \leftarrow f_{conv}(I^n, R_{out}, s)$
- 20          $I^n \leftarrow I^{new}$
- 21          $i \leftarrow i + 1$
- 22     **end**
- 23      $F^n \leftarrow f_{conv}(I, R_{out}, s)$  ▷ Define the score matrix.
- 24      $S \leftarrow F^n$  ▷ Build final score matrix.
- 25      $S_{model} \leftarrow \min(f_{err}(S_{r_i}, I_{class}))$
- 26 **end**
- 27 **return**  $S_{model}$  and  $R_{out}$



**Figure 7.** The architecture of the DeepCA feature extraction approach for breast cancer image classification.

### 3.3. DeepCA Training

To train DeepCA, the input images are randomly extracted into patches as mentioned above, where the RGB channels of each input image are processed separately. Then, each layer of DeepCA is trained according to the process represented in Figure 6 and Algorithm 1, as follows.

First, the Moore neighborhood and the rule vector are initialized based on the totalistic rule space. Each image patch is also defined as the current state for the CA neighborhood ( $S^t$ ); then, the next generation of image pixels ( $S^{t+1}$ ) is obtained via the convolution function ( $f_{conv}$ ) according to Equations (5), (7) and (10). The feature matrices  $F^n$  are generated from this process, where the depths of the feature matrices (or a number of  $f_{r_i}$  in  $F^n$ ) are determined according to the size of the rule vector  $R^n$  for  $f_{r_1}$  to  $f_{r_m}$  (see Equations (7) and (10)).

Second, the first score matrices are obtained by the feature matrices according to Equation (8). The *maxPool* function is employed at all layers of the feature matrices. Then, the score matrices can be obtained by combining the feature matrices derived from all layers (as the predicted objective map) as  $F_{max}^n$  (see Figure 7). After employing all images and rules in the rule vector  $R^n$ , we determine the rules  $r_i$  from the maximum value obtained by the summation function and register them as new members of the output rule vector  $R_{out}$ , defined as a set of rules. In addition, the new generations of the input images are also generated by the convolution function ( $f_{conv}$ ) with the output rule vector  $R_{out}$ , and then the same process is repeated to determine the next rule.

Finally, to build the final score matrices  $S$ , the convolution function ( $f_{conv}$ ) is applied to the input images with the output rule vector  $R_{out}$  obtained by the previous step, and then the summation function is applied. The final score matrices are then produced as the best score matrices of the model ( $S_{model}$ ) by determining the minimized error during the training process. In this regard, we adopt the mean squared error (MSE) function as a loss function to minimize the errors between the score matrices ( $S_{r_i}$ ) and their corresponding classes ( $I_{class}$ ) (or labelled data). We estimate an error value by using a loss function ( $f_{err}$ ) based on Equation (9). For the convergence criteria, the training process of each layer is repeated until all rules in the rule vector converge with training images or until they meet the limit of the desired number of the rule. Then, the best final score matrices of the model and the rule vector are applied in the FCNN training process. We then evaluate the classification performance based on the number of layers of DeepCA to determine whether to add more suitable layers.

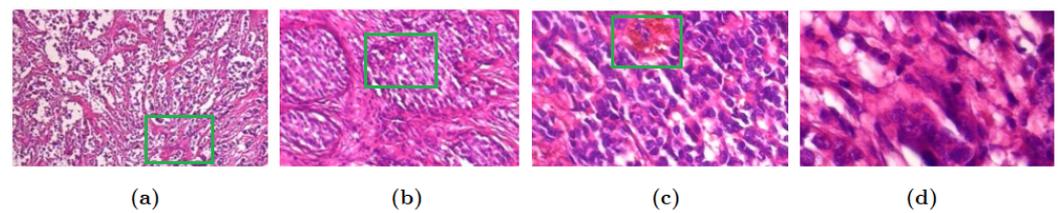
## 4. Experiments and Results

This section provides details about the utilized dataset and the configuration of the classifier, and then the classification performance of the proposed methods is evaluated.

### 4.1. Dataset

In this paper, we implement the proposed methods on the popular BreCaKHis dataset [5]. It is a publicly available breast cancer image dataset consisting of 7,909 microscopic images divided into two main classes (benign and malignant) obtained from 82 anonymous patients at various magnification factors. The original images are captured in the standard RGB format with a total of  $752 \times 582$  pixels at different magnifying factors:  $40\times$ ,  $100\times$ ,  $200\times$  and  $400\times$  (see Figure 8). Without compression or normalization, all images are cropped to  $700 \times 460$  pixels to remove black border areas and saved in the portable network graphics format.

The BreCaKHis dataset was gathered during a clinical study in 2014 with approval from the relevant authorities; the patients were invited to participate and were then referred to the P&D Laboratory (Brazil) to collect samples. Hematoxylin and eosin (HE) stained breast tissue biopsy slides were used to create these samples. Surgical open biopsy (SOB) or a surgical procedure via a small incision of human skin was used to collect these samples, which were then labeled by pathologists at the P&D Lab. Additionally, expert pathologists and additional tests, such as immunohistochemistry diagnosis, were used to confirm all sample cases. The dataset's picture distribution is shown in Table 1.



**Figure 8.** Slides of malignant breast tumors at different magnification factors: (a) 40×, (b) 100×, (c) 200×, and (d) 400×. The green rectangle is an interesting area that is detailed in slide obtained at the next magnification factor.

**Table 1.** Image distributions by their magnification factors and classes.

Magnification	Benign	Malignant	Total
40×	625	1370	1995
100×	644	1437	2081
200×	623	1390	2013
400×	588	1232	1820
Total	2480	5429	7909
Patients	24	58	82

**Table 2.** Three configurations of the FCNN classifiers that are used in the proposed methods.

Methods	Size of the Input Layer	No. and Sizes of Fully Connected Layers	Size of the Output Layer
Proposed method 1	12,288	1:64	2
Proposed method 2	12,288	2:64, 64	2
Proposed method 3	12,288	3:64, 64, 32	2

#### 4.2. Training Parameters of the Classifier

To confirm the performance of the DeepCA feature extraction approach, we adopt DeepCA in combination with FCNNs. In this regard, the input layer of each FCNN is defined as  $64 \times 64 \times 3$  or 12,888 neurons according to the image channels. In the experiment, we combine DeepCA with three types of FCNNs, in which the network configurations are defined as (64), (64, 64), and (64, 64, 32) (see Table 2). In addition, we implement stochastic gradient descent with momentum (SGDM) [63] as an optimizer for training the FCNNs with the following parameters: the number of dataset training iterations is set to 30 epochs, the minibatch size is set to 1, the momentum is defined as 0.9, and the learning rate is set to 0.000001.

#### 4.3. Performance Evaluation

To evaluate the performance of the proposed method against the state-of-the-art algorithms, the same experimental protocol used in the corresponding works is adopted to conduct an impartial comparison. More specifically, the image classification rate at the image level is estimated as:

$$\text{Image Classification Rate} = \frac{N_{\text{correct}}}{N_{\text{all}}} \quad (13)$$

where  $N_{\text{correct}}$  denotes the number of images that are correctly classified at each magnification factor and  $N_{\text{all}}$  denotes the total number of cancer images in the test set.

In addition, the model's performance when the training data set is unbalanced was evaluated using precision, recall, and F-Score. The metrics' formulations are explained as follows:

$$Precision = \frac{true\ positive}{true\ positive + false\ positive} \tag{14}$$

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \tag{15}$$

$$F - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{16}$$

where the number of malignant samples that were successfully identified as malignant is represented by *true positive*, and the number of benign samples that were correctly identified as benign is represented by *true negative*. Additionally, the *false positive* and *false negative* values show the number of benign samples that were mistakenly categorized as malignant and malignant samples that were mistakenly classed as benign, respectively.

#### 4.4. Results

To evaluate the performance of the proposed methods, we use the BreakHis dataset for a classification case with two classes: benign and malignant. Three different configurations of the proposed methods produce different classification results, as shown in Tables 3 and 4. We compare the classification accuracies of the proposed methods with the classification accuracy reports obtained both with variance and without any variance (dependent on the authors). However, in the case of the accuracies reported with variance, these values are obtained by computing the average rate over five trials according to the experimental protocol of [5,37,64]. We integrate DeepCA with the three FCNN structures defined in Table 2 as proposed method 1, proposed method 2, and proposed method 3. The classification results obtained at all magnification factors are presented in Tables 3 and 4 and Figures 9–12. Table 3 compares the classification performances of several feature extraction or feature description methods. The classification performances achieved at different magnification factors are obtained for all classes of images at each magnification factor contained in the testing set. Table 4 compares the best classification accuracies and improvement obtained by the proposed method and the state-of-the-art methods. The experimental results show that the proposed method achieves better classification accuracies between 96.0% to 97.2% for the proposed method 3, and achieves better results up to 7.95% improvement on average when compared with the state-of-the-art methods. In addition, Table 5 compares the precision, recall, and F-Score obtained by the proposed method and the state-of-the-art methods. The experimental results show that the proposed method achieves high precision, recall, and F-Score at 95.4%, 96.2%, and 95.6%, respectively.

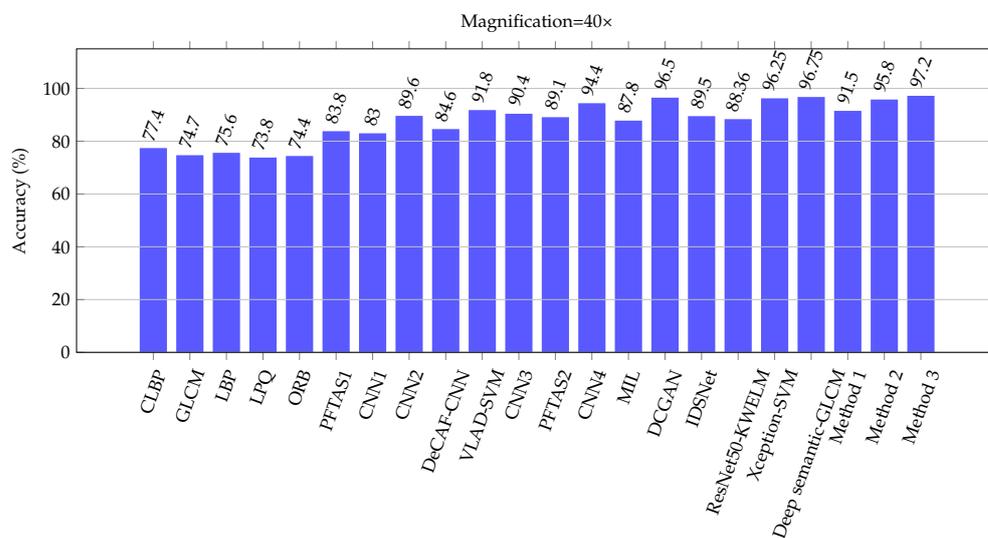


Figure 9. Comparison among the classification accuracy achieved by the proposed method and the state-of-the-art methods at a magnification factor of 40x.

**Table 3.** Comparison among the classification accuracy achieved with different methods.

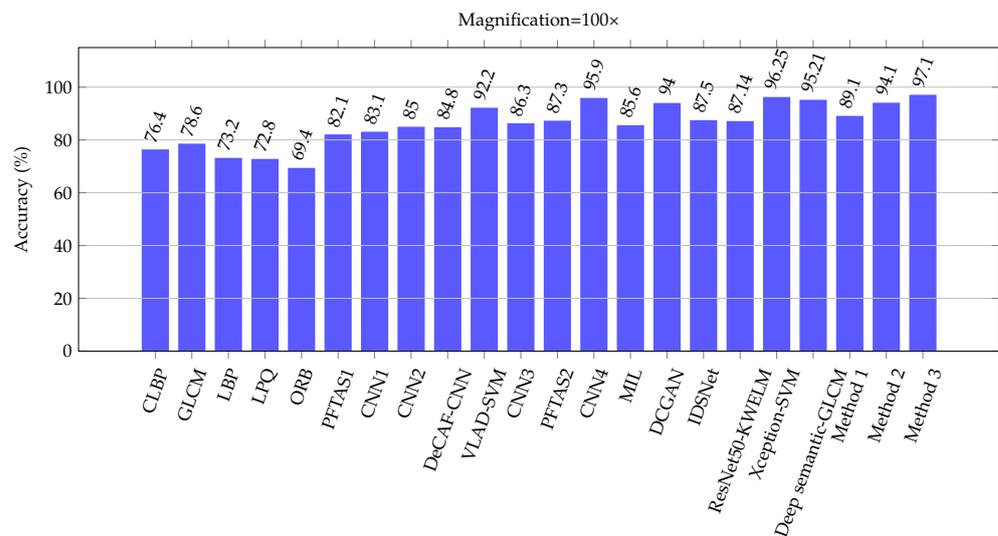
Methods	Authors	Magnification Factors				Average
		40×	100×	200×	400×	
CLBP	[5]	77.4	76.4	70.2	72.8	74.20
GLCM	[5]	74.7	78.6	83.4	81.7	79.60
LBP	[5]	75.6	73.2	72.9	73.1	73.70
LPQ	[5]	73.8	72.8	74.3	73.7	73.65
ORB	[5]	74.4	69.4	69.6	67.6	70.25
PFTAS1	[5]	83.8	82.1	85.1	82.3	83.33
CNN1	[65]	83.1	83.2	84.6	82.1	83.20
CNN2	[64]	89.6	85.0	84.0	80.8	84.85
DeCAF-CNN	[29]	84.6	84.8	84.2	81.6	83.80
VLAD-SVM	[24]	91.8	92.2	91.6	90.5	91.53
CNN3	[66]	90.4	86.3	83.1	81.3	85.28
PFTAS2	[25]	89.1	87.3	91.0	86.6	88.50
CNN4	[67]	94.4	95.9	<b>97.1</b>	<b>96.0</b>	95.85
MIL	[37]	87.8	85.6	81.7	82.9	84.50
DCGAN	[38]	96.5	94.0	95.5	93.0	94.75
IDSNet	[40]	89.5	87.5	90.0	84.0	87.75
ResNet50-KWELM	[39]	88.36	87.14	90.02	84.16	87.42
Xception-SVM	[41]	96.25	96.25	95.74	94.11	95.59
Deep semantic-GLCM	[42]	96.75	95.21	96.57	93.15	95.42
Proposed method 1		91.5	89.1	88.8	87.4	89.20
Proposed method 2		95.8	94.1	94.9	95.3	95.03
Proposed method 3		<b>97.2</b>	<b>97.1</b>	96.3	<b>96.0</b>	<b>96.65</b>

**Table 4.** Comparison between the best classification accuracy obtained by the proposed method and those of the state-of-the-art methods.

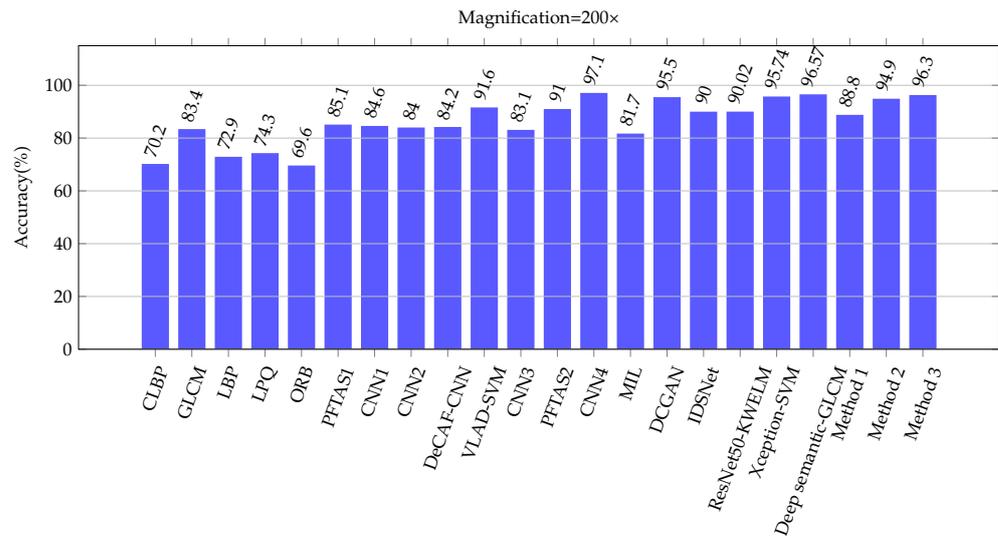
Methods	Authors	Magnification Factors				Average	Improvement		
		40×	100×	200×	400×		Proposed Method 1	Proposed Method 2	Proposed Method 3
PFTAS1	[5]	83.8 ± 4.1	82.1 ± 4.9	85.1 ± 3.1	82.3 ± 3.8	83.33	5.87	11.70	13.32
CNN1	[65]	83.08 ± 2.08	83.17 ± 3.51	84.63 ± 2.72	82.10 ± 4.42	83.25	5.95	11.78	13.40
CNN2	[64]	89.6 ± 6.5	85.0 ± 4.8	84.0 ± 3.2	80.80 ± 3.1	84.80	4.35	10.18	11.80
DeCAF-CNN	[29]	84.6 ± 2.9	84.8 ± 4.2	84.2 ± 1.7	81.6 ± 3.7	83.80	5.40	11.23	12.85
VLAD-SVM	[24]	91.8	92.2	91.6	90.5	91.53	-2.33	3.50	5.12
CNN3	[66]	90.4 ± 1.5	86.3 ± 3.3	83.1 ± 2.2	81.3 ± 3.5	85.28	3.92	9.75	11.37
PFTAS2	[25]	89.1	87.3	91.0	86.6	88.50	0.70	6.53	8.15
CNN4	[67]	94.4	95.9	<b>97.1</b>	<b>96.0</b>	95.85	-6.65	-0.82	0.80
MIL	[37]	87.8 ± 5.6	85.6 ± 4.3	81.7 ± 4.4	82.9 ± 4.1	84.50	4.70	10.53	12.15
DCGAN	[38]	96.5	94.0	95.5	93.0	94.75	-5.55	0.28	1.90
IDSNet	[40]	89.5 ± 2.0	87.5 ± 2.9	90.0 ± 5.3	84.0 ± 2.9	87.75	1.45	7.28	8.9
ResNet50-KWELM	[39]	88.36	87.14	90.02	84.16	87.42	1.78	7.61	9.23
Xception-SVM	[41]	96.25	96.25	95.74	94.11	95.59	-6.39	-0.56	1.06
Deep semantic-GLCM	[42]	96.75 ± 1.96	95.21 ± 2.18	96.57 ± 1.82	93.15 ± 2.30	95.42	-6.22	-0.4	1.23
Proposed method 1		91.5 ± 2.29	89.1 ± 3.9	88.8 ± 4.36	87.4 ± 4.77	89.20	-	-	-
Proposed method 2		95.8 ± 0.39	94.1 ± 0.45	94.9 ± 0.3	95.3 ± 0.34	95.03	-	-	-
Proposed method 3		<b>97.2 ± 0.13</b>	<b>97.1 ± 0.2</b>	96.3 ± 0.18	<b>96.0 ± 0.23</b>	<b>96.65</b>	-	-	-
Improvement Average							<b>0.5</b>	<b>6.33</b>	<b>7.95</b>

**Table 5.** Comparison between the precision, recall, and F-Score obtained by the proposed method and the state-of-the-art methods.

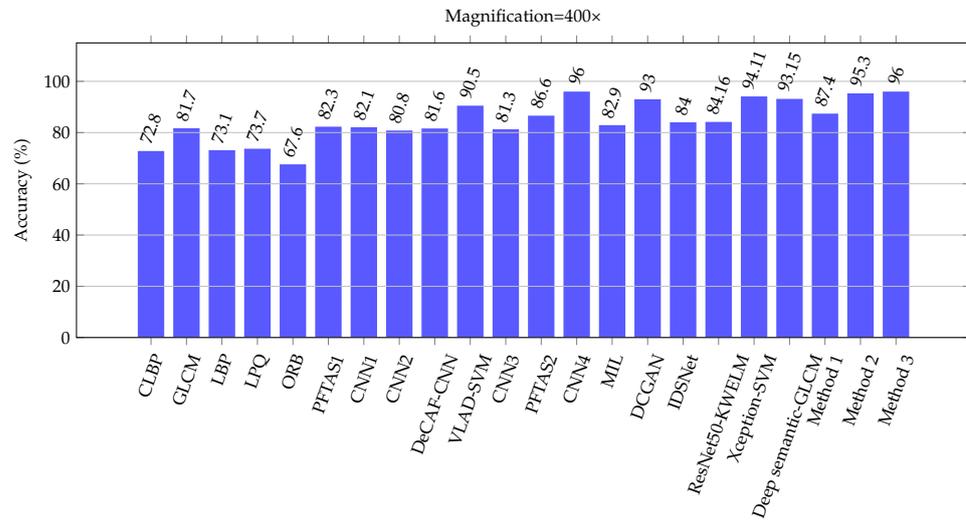
Methods	Magnifications	Precision	Recall	F-Score
ResNet50 + KWELM [39]	40×	87.2	86.2	86.6
	100×	85.2	88.0	86.1
	200×	88.6	89.2	88.7
	400×	82.1	84.2	82.8
Xception-SVM [41]	40×	96.0	96.0	96.0
	100×	96.0	96.0	96.0
	200×	95.0	95.0	95.0
	400×	95.0	93.0	93.0
Deep semantic-GLCM [42]	40×	97.5	96.9	97.2
	100×	97.5	96.0	96.7
	200×	97.9	97.4	97.7
	400×	96.4	94.0	95.1
Proposed method 1	40×	89.3	84.5	86.8
	100×	87.7	79.2	83.3
	200×	88.0	78.5	83.0
	400×	78.4	81.2	81.2
Proposed method 2	40×	93.0	93.9	93.4
	100×	91.3	89.8	90.5
	200×	92.3	91.4	91.9
	400×	92.1	92.7	92.7
Proposed method 3	40×	95.4	95.8	95.6
	100×	94.4	96.2	95.3
	200×	95.2	93.2	94.2
	400×	94.2	93.9	93.9



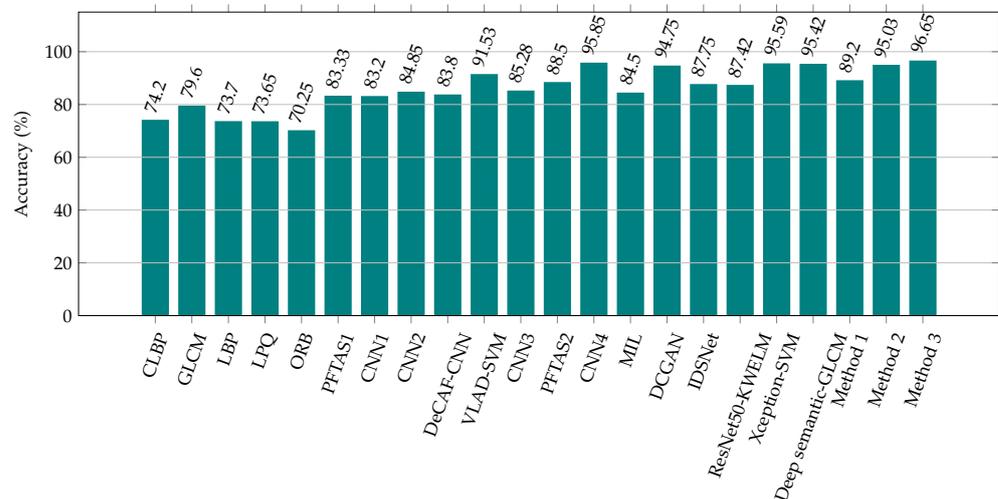
**Figure 10.** Comparison among the classification accuracy achieved by the proposed method and the state-of-the-art methods at a magnification factor of 100×.



**Figure 11.** Comparison among the classification accuracy achieved by the proposed method and the state-of-the-art methods at a magnification factor of 200x.



**Figure 12.** Comparison among the classification accuracy achieved by the proposed method and the state-of-the-art methods at a magnification factor of 400x.



**Figure 13.** Comparison among the average image classification accuracy achieved by the proposed method and the state-of-the-art methods at all magnification factors.

#### 4.5. Discussions

We compare the proposed method with various state-of-the-art models, i.e., the CLBP [5], GLCM [5], LBP [5], LPQ [5], ORB [5], PFTAS [5,25], VLAD-SVM [24], and CNN approaches in various architectures, i.e., [29,64–67], MIL [37], DCGAN [38], IDSNet [40], ResNet50-KWELM [39], Xception-SVM [41], and Deep semantic-GLCM [42]. It is found that the proposed DeepCA feature extraction technique combined with the FCNN classifiers work significantly better than other traditional approaches and the state-of-the-art methods for images at all magnification factors; this is especially true for proposed method 3. In this regard, the classification accuracy obtained by proposed method 3 for the 40× magnification factor is the highest at 97.2%, followed by those of the Deep semantic-GLCM [42], DCGAN [38] and proposed method 2, which achieve accuracies of 96.75%, 96.5% and 95.8%, respectively. For the 100× magnification factor, proposed method 3 also obtains the highest classification accuracy (97.1%) followed by the Xception-SVM [41] (96.25%) and CNN4 proposed by [67] (95.9%) whereas proposed method 2 and the DCGAN [38] achieve approximate accuracy rates of 94.1% and 94.0%, respectively. Unfortunately, in the case of the 200× magnification factor, the proposed methods cannot achieve the best classification accuracy. Proposed method 1, proposed method 2, and proposed method 3 only produce accuracies of 88.8%, 94.9%, and 96.3%, respectively. In contrast, the highest classification accuracy of 97.1% is achieved by the CNN4 followed by Deep semantic-GLCM [42] which achieve accuracies of 96.57%. However, the classification accuracies obtained by the proposed methods are still better than those of several traditional methods. At the 400× magnification factor, proposed method 3 and the CNN4 also achieve the highest classification accuracy (96.0%) followed by proposed method 2 (95.3%). As seen in Table 3, although the classification accuracy of the CNN4 is quite consistently better for 40×, 100× and 200× magnified images as compared to CNN1 [65], CNN2 [64], DeCAF-CNN [29], CNN3 [66], and the proposed method 3. However, the performance of the proposed methods is obtained by all automatic feature extraction processes, whereas CNN4 combined the CNN with several hand-crafted feature extraction methods. Moreover, Figure 13 illustrates a comparison between the average classification efficiencies of the state-of-the-art approaches and the proposed methods at all magnification factors. It can be seen that the proposed methods, especially proposed method 3, can significantly and effectively classify the input images.

#### 4.6. Parameter Sensitivity

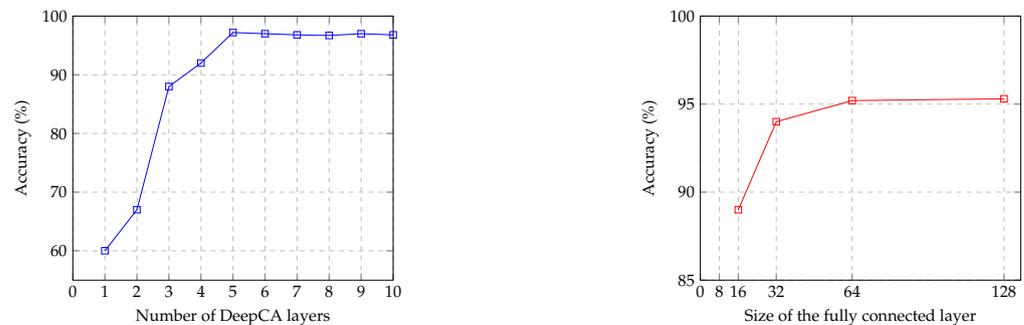
We attempt to identify the model parameters in our experiment to determine the optimal model. In this section, we demonstrate how different model parameters affect the resulting model performance.

**Impact of the number of DeepCA layers.** The number of layers is a significant parameter in the DeepCA structure that directly affects the model performance. Each layer directly executes with all channels of the input image and the rule vector that aims to tune the data values for classification. Therefore, if the number of layers is too small, the data tuned by the model are insufficient for efficient classification. On the other hand, if the number of layers is too large, the data tuned by the model can be classified. However, this barely increases the classification accuracy, and the required computation time is too long. In this regard, we test the layer number parameter from 1 to 10. Figure 14 shows the impact of the number of layers on the BreakHis dataset in terms of classification accuracy. The appropriate number of layers chosen in the experiment is five based on the maximal achieved classification accuracy.

**Impact of the size of the fully connected layer.** This parameter is related to the interconnection within the fully connected layer of each FCNN. Due to the performance of the FCNNs, even if the fully connected layers have small sizes, e.g., 16, they are sufficient for small classification tasks. However, larger fully connected layers tend to yield better classification performance. As shown in Figure 14, we vary the size of the fully connected layer to test the resulting classification performance. In this regard, we fix the number of DeepCA layers to five layers while keeping other parameters fixed in the FCNNs as

follows: the number of dataset training iterations is set to 30 epochs, the minibatch size is set to 1, the momentum is defined as 0.9, and the learning rate is set to 0.000001.

From the experimental results, the fully connected layer size of 64 is chosen as the best choice for further use. However, we extend the experiment by combining several FCNN sizes to achieve better classification performance (see Table 2), as shown in the performance report of this paper.



**Figure 14.** Impacts of the number of layers and the size of the fully connected layer on the resulting classification accuracy.

## 5. Conclusions

In this paper, a novel deep feature extraction method based on CA theory for image classification is proposed. The proposed framework relies on multilayer CA and a deep learning approach. It is divided into two main parts: deep feature extraction, which relies on multilayer CA, and evolution rules. The first part aims to extract the multilayer features (or feature matrices) of images, which are called score matrices. The score matrices are then applied in the subsequent decision stage. These score matrices are flattened before being fed into the fully connected layer of an ANN for the training phase and used for classification during the test phase. Finally, histopathological images from the BreakHis dataset, the benchmark breast cancer image dataset, are deeply applied to confirm the efficiency of the proposed method. In addition, we present three model configurations to explore the best classification accuracies of the proposed method. The empirical results show that the proposed method achieves a classification accuracy of 97.2% at a 40× magnification factor, which is better than that of the compared state-of-the-art methods. In addition, the experimental results show that the proposed method achieves the better results up to 9.09% improvement on average when compared with the state-of-the-art methods.

According to a limitation of the computing facility, we implement our proposal in the existing available computing environment that still exhibits the outstanding results. However, there are some limitations to this work. Despite applying the CA with the rules generated many variation data, the proposed model still has the opportunity to improve its classification performance on larger datasets because the limited size of data in the dataset affects classification accuracy. For future research, we aim to expand more layers of cellular automata to improve the feature extraction performance and plan to design a complete framework that relies on a pure CA approach for both feature learning and classification engine. Besides, we plan to extend the model to the larger classification problem, i.e., subtypes of benign and malignant classes of breast cancer images or others images datasets, by redesigning the new structure of CA feature extraction and learning algorithm. Although the proper evolution rules of CA on the larger datasets are hard to determine, it still has room for improvement. In this regard, the optimization techniques can be applied to determine the suitable evolution rules of CA. For computation purposes, the same as denser CNN, high-performance computing with high-speed GPU and huge memory is necessary.

**Author Contributions:** Conceptualization, S.T. and S.W.; methodology, S.T. and S.W.; software, S.T.; validation, S.T. and S.W.; formal analysis, S.T.; investigation, S.T.; resources, S.T.; data curation, S.T.; writing—original draft preparation, S.T. and S.W.; writing—review and editing, S.T. and S.W.; visualization, S.T.; supervision, S.W.; project administration, S.T.; funding acquisition, S.T. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data that support the findings of this study are openly available in the Federal University of Parana (UFPR) repository at <https://web.inf.ufpr.br/vri/databases> (accessed on 29 March 2021).

**Acknowledgments:** This work is supported by the Artificial Intelligence Center (AIC), MLIS Laboratory, College of Computing, Khon Kaen University, Thailand.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

1-NN	1-nearest neighbor
ANN	Artificial Neural Network
CA	Cellular Automata
CAT	Computer-Aided Tomography
CLBPs	Completed local binary patterns
CNN	Convolutional Neural Networks
CSAE	Convolutional sparse autoencoder
DBT	Digital breast tomosynthesis
DCNN	Deep convolutional neural networks
DeepCA	Deep cellular automata
ECA	Elementary cellular automaton
FCNNs	Fully connected neural networks
GLCMs	Grey-level co-occurrence matrices
LBPs	Local binary patterns
MIL	Multiple-instance learning
MLP	Multilayer perceptrons (MLP)
MRI	Magnetic Resonance Imaging
PFTASs	Threshold adjacency statistics
QDA	Quadratic linear analysis
RF	Random forest
SSAE	Stacked sparse autoencoder
SVMs	Support vector machines
VLAD	Vector of locally aggregated descriptors

## References

1. Skandalakis, J.E. Embryology and anatomy of the breast. In *Breast Augmentation*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 3–24. [\[CrossRef\]](#)
2. Ellis, H.; Colborn, G.L.; Skandalakis, J.E. Surgical embryology and anatomy of the breast and its related anatomic structures. *Surg. Clin. North Am.* **1993**, *73*, 611. [\[CrossRef\]](#) [\[PubMed\]](#)
3. Rubin, R.; Strayer, D.S.; Rubin, E. *Rubin's Pathology: Clinicopathologic Foundations of Medicine*; Lippincott Williams & Wilkins: Philadelphia, PA, USA, 2008.
4. Lakhani, S.R.; Ellis, I.O.; Schnitt, S.; Tan, P.H.; van de Vijver, M. *WHO Classification of Tumours of the Breast*; IARC: Lyon, France, 2012.
5. Spanhol, F.A.; Oliveira, L.S.; Petitjean, C.; Heutte, L. A dataset for breast cancer histopathological image classification. *IEEE Trans. Biomed. Eng.* **2015**, *63*, 1455–1462. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Haralick, R.M.; Shanmugam, K.; Dinstein, I.H. Textural features for image classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *6*, 610–621. [\[CrossRef\]](#)

7. Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 971–987. [[CrossRef](#)]
8. Hamilton, N.A.; Pantelic, R.S.; Hanson, K.; Teasdale, R.D. Fast automated cell phenotype image classification. *BMC Bioinform.* **2007**, *8*, 110. [[CrossRef](#)]
9. Ojansivu, V.; Heikkilä, J. Blur insensitive texture classification using local phase quantization. In Proceedings of the International Conference on Image and Signal Processing, Cherbourg-Octeville, France, 1–3 July 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 236–243. [[CrossRef](#)]
10. Guo, Z.; Zhang, L.; Zhang, D. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process.* **2010**, *19*, 1657–1663. [[CrossRef](#)]
11. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 2564–2571. [[CrossRef](#)]
12. Agarwal, N.; Ford, K.H.; Shneider, M. Sentence boundary detection using a maxEnt classifier. *Proc. MISC* **2005**, 1–6.
13. Mohammed, N.F.; Omar, N. Arabic named entity recognition using artificial neural network. *J. Comput. Sci.* **2012**, *8*, 1285.
14. Perboli, G.; Gajetti, M.; Fedorov, S.; Giudice, S.L. Natural language processing for the identification of human factors in aviation accidents causes: An application to the SHEL methodology. *Expert Syst. Appl.* **2021**, *186*, 115694. [[CrossRef](#)]
15. Sompong, C.; Wongthanavas, S. An efficient brain tumor segmentation based on cellular automata and improved tumor-cut algorithm. *Expert Syst. Appl.* **2017**, *72*, 231–244. [[CrossRef](#)]
16. Zhang, Z.; Wu, Z.; Jiang, Q.; Du, L.; Hu, L. Co-saliency Detection Based on Superpixel Matching and Cellular Automata. *TIIS* **2017**, *11*, 2576–2589. [[CrossRef](#)]
17. Liu, Y.; Yuan, P. Saliency Detection Using Global and Local Information Under Multilayer Cellular Automata. *IEEE Access* **2019**, *7*, 72736–72748. [[CrossRef](#)]
18. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, San Francisco, CA, USA, 30 November–3 December 1992; pp. 1097–1105.
20. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
21. Nichele, S.; Molund, A. Deep learning with cellular automaton-based reservoir computing. *Complex Systems* **2017**, *26*, 319–340. [[CrossRef](#)]
22. Tangsakul, S.; Wongthanavas, S. Single Image Haze Removal Using Deep Cellular Automata Learning. *IEEE Access* **2020**, *8*, 103181–103199. [[CrossRef](#)]
23. Zhang, B. Breast cancer diagnosis from biopsy images by serial fusion of Random Subspace ensembles. In Proceedings of the 2011 4th International Conference on Biomedical Engineering and Informatics (BMEI), Shanghai, China, 15–17 October 2011; Volume 1, pp. 180–186.
24. Dimitropoulos, K.; Barmpoutis, P.; Zioga, C.; Kamas, A.; Patsiaoura, K.; Grammalidis, N. Grading of invasive breast carcinoma through Grassmannian VLAD encoding. *PLoS ONE* **2017**, *12*, e0185110. [[CrossRef](#)]
25. Alirezazadeh, P.; Hejrati, B.; Monsef-Esfahani, A.; Fathi, A. Representation learning-based unsupervised domain adaptation for classification of breast cancer histopathology images. *Biocybern. Biomed. Eng.* **2018**, *38*, 671–683. [[CrossRef](#)]
26. Albarqouni, S.; Baur, C.; Achilles, F.; Belagiannis, V.; Demirci, S.; Navab, N. Aggnet: deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE Trans. Med. Imaging* **2016**, *35*, 1313–1321. [[CrossRef](#)]
27. Suzuki, S.; Zhang, X.; Homma, N.; Ichiji, K.; Sugita, N.; Kawasumi, Y.; Ishibashi, T.; Yoshizawa, M. Mass detection using deep convolutional neural network for mammographic computer-aided diagnosis. In Proceedings of the 2016 55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Tsukuba, Japan, 20–23 September 2016; pp. 1382–1386.
28. Ertosun, M.G.; Rubin, D.L. Probabilistic visual search for masses within mammography images using deep learning. In Proceedings of the 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Washington, DC, USA, 9–12 November 2015; pp. 1310–1315. [[CrossRef](#)]
29. Spanhol, F.A.; Oliveira, L.S.; Cavalin, P.R.; Petitjean, C.; Heutte, L. Deep features for breast cancer histopathological image classification. In Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, 5–8 October 2017; pp. 1868–1873. [[CrossRef](#)]
30. Xu, J.; Xiang, L.; Hang, R.; Wu, J. Stacked Sparse Autoencoder (SSAE) based framework for nuclei patch classification on breast cancer histopathology. In Proceedings of the 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), Beijing, China, 29 April–2 May 2014; pp. 999–1002. [[CrossRef](#)]
31. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the 22nd ACM International Conference on Multimedia, Orlando, FL, USA, 3–7 November 2014; pp. 675–678. [[CrossRef](#)]
32. Dhungel, N.; Carneiro, G.; Bradley, A.P. Deep structured learning for mass segmentation from mammograms. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 2950–2954. [[CrossRef](#)]
33. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255. [[CrossRef](#)]

34. Kallenberg, M.; Petersen, K.; Nielsen, M.; Ng, A.Y.; Diao, P.; Igel, C.; Vachon, C.M.; Holland, K.; Winkel, R.R.; Karssemeijer, N.; et al. Unsupervised deep learning applied to breast density segmentation and mammographic risk scoring. *IEEE Trans. Med. Imaging* **2016**, *35*, 1322–1331. [[CrossRef](#)]
35. Kim, D.H.; Kim, S.T.; Ro, Y.M. Latent feature representation with 3-D multi-view deep convolutional neural network for bilateral analysis in digital breast tomosynthesis. In Proceedings of the 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 927–931. [[CrossRef](#)]
36. Swiderski, B.; Kurek, J.; Osowski, S.; Kruk, M.; Barhoumi, W. Deep learning and non-negative matrix factorization in recognition of mammograms. In Proceedings of the Eighth International Conference on Graphic and Image Processing (ICGIP 2016), Tokyo, Japan, 29–31 October 2016; International Society for Optics and Photonics: Bellingham, WA, USA, 2017; Volume 10225, p. 102250B. [[CrossRef](#)]
37. Sudharshan, P.; Petitjean, C.; Spanhol, F.; Oliveira, L.E.; Heutte, L.; Honeine, P. Multiple instance learning for histopathological breast cancer image classification. *Expert Syst. Appl.* **2019**, *117*, 103–111. [[CrossRef](#)]
38. Saini, M.; Susan, S. Deep transfer with minority data augmentation for imbalanced breast cancer dataset. *Appl. Soft Comput.* **2020**, *97*, 106759. [[CrossRef](#)]
39. Saxena, S.; Shukla, S.; Gyanchandani, M. Breast cancer histopathology image classification using kernelized weighted extreme learning machine. *Int. J. Imaging Syst. Technol.* **2021**, *31*, 168–179. [[CrossRef](#)]
40. Li, X.; Shen, X.; Zhou, Y.; Wang, X.; Li, T.Q. Classification of breast cancer histopathological images using interleaved DenseNet with SENet (IDSNet). *PLoS ONE* **2020**, *15*, e0232127. [[CrossRef](#)]
41. Sharma, S.; Kumar, S. The Xception model: A potential feature extractor in breast cancer histology images classification. *ICT Express* **2022**, *8*, 101–108. [[CrossRef](#)]
42. Hao, Y.; Zhang, L.; Qiao, S.; Bai, Y.; Cheng, R.; Xue, H.; Hou, Y.; Zhang, W.; Zhang, G. Breast cancer histopathological images classification based on deep semantic features and gray level co-occurrence matrix. *PLoS ONE* **2022**, *17*, e0267955. [[CrossRef](#)]
43. Atban, F.; Ekinci, E.; Garip, Z. Traditional machine learning algorithms for breast cancer image classification with optimized deep features. *Biomed. Signal Process. Control.* **2023**, *81*, 104534. [[CrossRef](#)]
44. Neumann, J.; Burks, A.W. *Theory of self-reproducing automata*; University of Illinois Press Urbana: Champaign, IL, USA, 1966; Volume 1102024.
45. Ulam, S. Some ideas and prospects in biomathematics. *Annu. Rev. Biophys. Bioeng.* **1972**, *1*, 277–292. [[CrossRef](#)] [[PubMed](#)]
46. Wolfram, S. Computation theory of cellular automata. *Commun. Math. Phys.* **1984**, *96*, 15–57. [[CrossRef](#)]
47. Sahin, U.; Uguz, S.; Sahin, F. Salt and pepper noise filtering with fuzzy-cellular automata. *Comput. Electr. Eng.* **2014**, *40*, 59–69. [[CrossRef](#)]
48. Wongthanavas, S.; Sadananda, R. A CA-based edge operator and its performance evaluation. *J. Vis. Commun. Image Represent.* **2003**, *14*, 83–96. [[CrossRef](#)]
49. Kumar, T.; Sahoo, G. A novel method of edge detection using cellular automata. *Int. J. Comput. Appl.* **2010**, *9*, 38–44. [[CrossRef](#)]
50. Rosin, P.L.; Sun, X. Edge detection using cellular automata. In *Cellular Automata in Image Processing and Geometry*; Springer: Cham, Switzerland, 2014; pp. 85–103. [[CrossRef](#)]
51. Diwakar, M.; Patel, P.K.; Gupta, K. Cellular automata based edge-detection for brain tumor. In Proceedings of the 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Mysore, India, 22–25 August 2013; pp. 53–59. [[CrossRef](#)]
52. Tourtounis, D.; Mitianoudis, N.; Sirakoulis, G.C. Salt-n-pepper noise filtering using cellular automata. *arXiv* **2017**, arXiv:1708.05019.
53. Qadir, F.; Shoosha, I.Q. Cellular automata-based efficient method for the removal of high-density impulsive noise from digital images. *Int. J. Inf. Technol.* **2018**, *10*, 529–536. [[CrossRef](#)]
54. Priego, B.; Prieto, A.; Duro, R.J.; Chanussot, J. A cellular automata-based filtering approach to multi-temporal image denoising. *Expert Syst.* **2018**, *35*, e12235. [[CrossRef](#)]
55. Qin, Y.; Feng, M.; Lu, H.; Cottrell, G.W. Hierarchical cellular automata for visual saliency. *Int. J. Comput. Vis.* **2018**, *126*, 751–770. [[CrossRef](#)]
56. Liu, Y.; Chen, Y.; Han, B.; Zhang, Y.; Zhang, X.; Su, Y. Fully automatic Breast ultrasound image segmentation based on fuzzy cellular automata framework. *Biomed. Signal Process. Control.* **2018**, *40*, 433–442. [[CrossRef](#)]
57. Li, C.; Liu, L.; Sun, X.; Zhao, J.; Yin, J. Image segmentation based on fuzzy clustering with cellular automata and features weighting. *EURASIP J. Image Video Process.* **2019**, *2019*, 1–11. [[CrossRef](#)]
58. Wali, A.; Saeed, M. Biologically inspired cellular automata learning and prediction model for handwritten pattern recognition. *Biol. Inspired Cogn. Archit.* **2018**, *24*, 77–86. [[CrossRef](#)]
59. Packard, N.H.; Wolfram, S. Two-dimensional cellular automata. *J. Stat. Phys.* **1985**, *38*, 901–946. [[CrossRef](#)]
60. Khan, A.R.; Choudhury, P.P.; Dihidar, K.; Mitra, S.; Sarkar, P. VLSI architecture of a cellular automata machine. *Comput. Math. Appl.* **1997**, *33*, 79–94. [[CrossRef](#)]
61. Uguz, S.; Akin, H.; Siap, I.; Sahin, U. On the irreversibility of Moore cellular automata over the ternary field and image application. *Appl. Math. Model.* **2016**, *40*, 8017–8032. [[CrossRef](#)]
62. Jana, B.; Pal, P.; Bhaumik, J. New image noise reduction schemes based on cellular automata. *Int. J. Soft Comput. Eng.* **2012**, *2*, 98–103.

63. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
64. Spanhol, F.A.; Oliveira, L.S.; Petitjean, C.; Heutte, L. Breast cancer histopathological image classification using convolutional neural networks. In Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, Canada, 24–29 July 2016; pp. 2560–2567. [[CrossRef](#)]
65. Bayramoglu, N.; Kannala, J.; Heikkilä, J. Deep learning for magnification independent breast cancer histopathology image classification. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 2440–2445. [[CrossRef](#)]
66. Mehra, R. Automatic magnification independent classification of breast cancer tissue in histological images using deep convolutional neural network. In Proceedings of the International Conference on Advanced Informatics for Computing Research, Shimla, India, 14–15 July 2018; Springer: Singapore, 2018; pp. 772–781. [[CrossRef](#)]
67. Nahid, A.A.; Kong, Y. Histopathological breast-image classification using local and frequency domains by convolutional neural network. *Information* **2018**, *9*, 19. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.