Deep Learning Techniques for Breast Cancer Detection Using Medical Image Analysis

D. Selvathi and A. Aarthy Poornila

Abstract Breast cancer has the second highest mortality rate in women next to lung cancer. As per clinical statistics, 1 in every 8 women is diagnosed with breast cancer in their lifetime. However, periodic clinical checkups and self-tests help in early detection and thereby significantly increase the chances of survival. Invasive detection techniques cause rupture of the tumor, accelerating the spread of cancer to adjoining areas. Hence, there arises the need for a more robust, fast, accurate, and efficient noninvasive cancer detection system. In this work, an automated system is proposed for achieving error-free detection of breast cancer using mammogram. In this system, the deep learning techniques such as convolutional neural network, sparse autoencoder, and stacked sparse autoencoder are used. The performance of these techniques is analyzed and compared with the existing methods. From the analysis, it is observed that the stacked sparse autoencoder performs better compared to other methods.

Keywords Breast cancer · Mammogram · Deep learning techniques

1 Introduction

Cancer is one of the deadliest diseases faced by mankind. The chance of women dying from early-stage breast cancer is estimated to be about 3%. Statistics show that breast cancer constitutes about one-fourth of all the cancers affecting women, commonly affecting women above 40 years of age. Recent statistics show that there has been a steady decrease in the annual deaths from breast cancer among women, from 32.69 in 1991 to 24.00 in 2005 (per 100,000 populations). This decrease in cancer death rates reflects the improvements in early detection and treatment for

D. Selvathi (✉) · A. Aarthy Poornila
Mepco Schlenk Engineering College, Sivakasi, Tamil Nadu, India
e-mail: dselvathi@mepcoeng.ac.in

A. Aarthy Poornila
e-mail: aarthypoornila@gmail.com

J. Hemanth and V.E. Balas (eds.), *Biologically Rationalized Computing Techniques For Image Processing Applications*, Lecture Notes in Computational Vision and Biomechanics 25, DOI 10.1007/978-3-319-61316-1_8

breast cancer. Research is being done for breast cancer prevention, detection, and treatment, in order to increase the survival rate of patients all over the world.

Medical imaging for breast cancer can be used as a noninvasive method for looking inside the body and assisting the doctors in diagnosis and treatment. Islam et al. [1] have explored different breast cancer imaging techniques such as mammography, magnetic resonance imaging (MRI), and ultrasound for breast cancer diagnosis and compared their effectiveness, advantages, and disadvantages for detecting early-stage breast cancer. In [2], Sachin Prasad Na and Dana Houserkovaa evaluated the role of various modalities (mammogram, molecular breast imaging, ultrasound, MRI, PET, thermography, galactography, CT scan) used in breast cancer screening and diagnosis. Though initial detection of breast cancer can be done using any one of the available imaging modalities, malignancy cannot be confirmed by such images alone. So the treatment does not start until microscopic examination of tissue from the tumor is done to confirm its malignancy. Shyamala et al. [3] say that when such procedures are done, there is a high risk of tumor cells being seeded into the interstitial tissue fluid or veins. There is also a risk of dragging cells along the surgical incision or needle track leading to the possibility of increasing the spread of cancer through biopsy.

The field of medical image processing gains its importance in the need of accurate and efficient diagnosis of diseases over a short period of time. Since manual process is tedious, time-consuming, and impractical for large data, a need for automatic processing arises which can revolutionize modern medicine. In breast cancer detection using mammogram, the mammographic breast image is usually preprocessed to remove the pectoral muscle, as it biases the detection process. Thus, by removing the background and pectoral muscle regions from the mammogram, the search for abnormalities can be limited to the breast profile region alone. Jawad Nagi et al. proposed a technique for automated segmentation of mammogram, which uses morphological preprocessing and seeded region growing (SRG) algorithm [4] in order to remove the pectoral muscle region from the breast profile region in the mammogram as the presence of such regions can mislead the abnormality detection.

Detection of cancer tissues with higher pixel intensity than the other breast region is done easily. In case of dense breasts, the normal tissues have intensities similar to the tumor regions, and in such cases, it is necessary to identify tumor regions successfully. Classification of the tumor tissues in the breast into benign and malignant is a tedious work. Feature extraction is an important step in mammogram analysis. In conventional methods, handcrafted features are used to represent the content of images. The neural network emerged as an alternative approach to extract the best features automatically. Maxine Tan et al. have predicted the risk of a patient being affected by breast cancer in the near future, based on the analysis of feature variations (density features, structural features, and texture features) in a series of mammographic images [5]. Singh and Gupta [6] proposed a two-step breast cancer detection method. First, the malignant area is found using filtering and thresholding operations. Then, morphological operations are done to find the region boundary in the tumor patch. In [7], Mellisa Pratiwi et al. used gray-level co-occurrence matrix (GLCM)-based texture features and radial basis

function neural network (RBFNN) for breast cancer classification using mammogram. For comparative analysis, backpropagation neural network (BPNN) is used in this work. It is reported that RBFNN gives 94.29% benign and malignant classification accuracy which is 2% greater than BPNN. Kim et al. [8] classified spiculated malignant lesions and normal tissues using support vector machine (SVM) which used a set of region-based stellate features, achieving 97.9% classification performance on mini-MIAS database. In [9], Hussain et al. employed successive enhancement learning-based weighted support vector machine (SELwSVM) to classify masses and normal dense tissues with 96% accurate recognition rate, by exploiting structural properties represented by the extracted texture features.

Deep learning is a newly emerging area which exploits artificial intelligence and machine learning to learn features directly from the data, using multiple nonlinear processing layers. Deep learning models can achieve very high accuracy in image classification, on par with human-level performance. Li Deng and Dong Yu discussed various deep learning techniques, their classification into supervised, unsupervised, and hybrid deep networks depending on their architectures and applications such as computer vision, language modeling, text processing, multimodal learning, and information retrieval [10]. Schmidhuber, in [11], extensively explained about the credit assignment problem and classification of shallow and deep learners by the depth of their credit assignment paths. Comparative analysis of acyclic (Feedforward) and cyclic (Recurrent) neural networks showed that RNNs are the deepest and are more powerful than FNNs. This paper also has a detailed discussion on supervised, unsupervised, and reinforced learning. Martin Langkvist et al. explored the time-series problems in deep learning and unsupervised feature learning. İn this work, a solution of using temporal coherence or temporal pooling or hidden unit activation is suggested for high-dimensional, multivariate data [12]. Furthermore, a comparative study of various time-series problems such as stock prediction, speech recognition, motion capture, and psychological data has been done.

Guo et al. [13] reviewed 216 research papers describing about deep learning algorithms and categorized them into convolutional neural network (CNN), autoencoders, restricted boltzmann machine (RBM), and sparse coding. Deeper analysis of the applications such as computer vision shows that such algorithms outperform accuracy of human raters. Arevalo et al. in [14] used CNN to learn features from mammograms, which are subsequently fed into a classifier. CNN architecture takes the raw pixels of the image as input, to learn a set of nonlinear transformations that represent better image data. This also outperforms a set of handcrafted features. Ketan Sharma and B. Preet used convolutional neural network classifier in [15] to boost the classification performance in breast cancer detection. CNN performs better than the other classifiers in terms of accuracy and misclassification rate of mammograms. Autoencoders are used in learning features in an unsupervised fashion in deep networks. Baldi in [16] gave detailed explanations about the types and the complexity of both linear and nonlinear deep networks and also elucidated about Hebbian learning, clustering, and information theory.

Mammographic risk scoring is automated by computing breast cancer risk from the extracted mammographic features in [17] by Kersten Petersen et al. In this work, the features are learned using a convolutional sparse autoencoder (CSAE) which uses sparse autoencoders in a convolutional architecture. Classification is performed by a softmax classifier based on these extracted features. Both density segmentation and texture scoring can be done in this manner. Jun Xu et al. in [18] used SSAE in breast cancer detection using histopathological images. High-level features can be learned from images by stacking sparse autoencoders together. These features are extracted patchwise and are fed into a classifier for classification. This work inspects various deep learning techniques and justifies that the performance of SSAE is superior compared to all other techniques.

Mammography, which is the present "Gold Standard" for breast imaging, is widely used for screening and diagnosing breast cancer in women. The radiation exposure in case of mammogram imaging is very minimal and also cheaper compared to all other techniques. There are many publicly available mammogram databases. The mini-MIAS database is one such public database that is used in this work, where an attempt has been made to explore the human intelligence-based deep learning techniques for achieving error-free detection of breast cancer.

2 Proposed System

The proposed system uses human intelligence-based deep learning techniques for detecting breast cancer using mammogram imaging modality. The various steps involved in the proposed system are shown in Fig. 1.

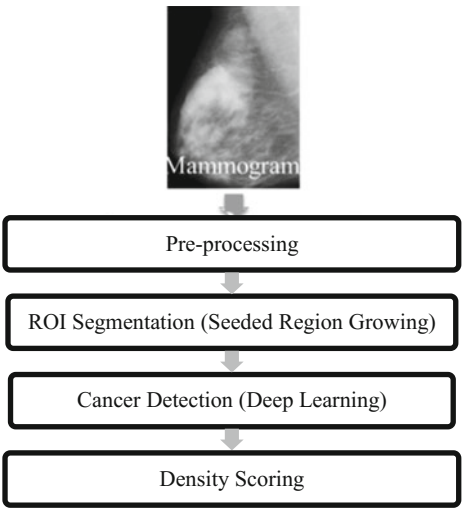


Fig. 1 Flow chart of proposed system

2.1 Input Imaging Modalities

The selection of a specific modality among all the medical imaging modalities available to study the organ present inside the human body is important. This will provide a wealth of information to understand the nature of diseases. The proposed system uses X-ray images (mammogram) which generate the information about the anatomical structure of a lesion.

2.2 Preprocessing

Image preprocessing enhances some important image features for further processing and suppresses unwanted distortions. Figure 2 shows the various stages of preprocessing method.

Mammogram usually suffers from digitization noises such as straight lines and radio-opaque artifacts in the form of labels and wedges. In the computer-aided detection process, identification of the pectoral muscle region and breast profile extraction is an essential step in the methodology of preprocessing.

Median filter and other morphological process are involved in the preprocessing step for the removal of noise and radio-opaque artifacts, respectively.

2.3 ROI Segmentation

An ROI is a certain region or part that has to be filtered out from the original image in order to perform further operations on it. This region can be extracted by multiplying the original image with a generated binary mask, which is of the same size as that of the image to be processed. The steps involved in ROI segmentation are given in Fig. 3.

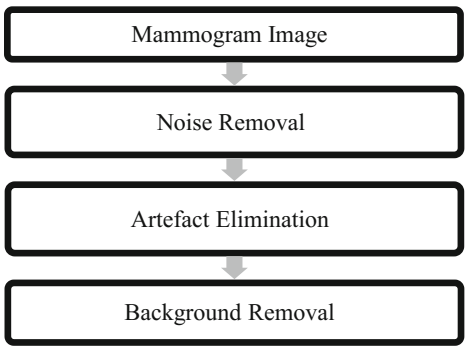


Fig. 2 Preprocessing method

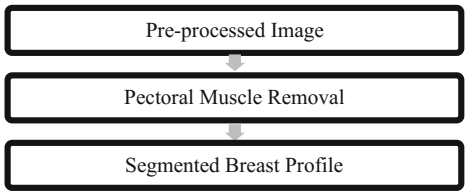


Fig. 3 ROI segmentation

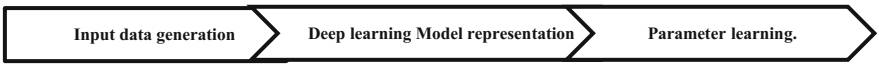


Fig. 4 Overall proposed approach

Region-based segmentation method such as seeded region growing is used for pectoral muscle removal. This approach of segmentation involves the initial seed point selection and determination of the neighboring pixels to be added to the region.

2.4 Cancer Detection by Deep Learning Technique

The architectures of "human-brain-inspired" deep nonlinear models compose complex features in the deeper layers of the network by analyzing the simple features learned in the previous layers. These features prove to be very effective descriptors in object recognition problems. During the training phase of these models, the features are encoded iteratively, and then, the learned weights are updated for improved optimization of the network. The features can be learned either using CNN in a supervised manner or using autoencoders in an unsupervised way. The features learned in a layerwise method are fed into a trained classifier, which predicts the labels. The classifier being a supervised layer has to be trained using a set of training images along with the associated label. The trained network should be able to accurately predict the label for an unseen image.

In this work, a deep convolutional neural network is used for supervised learning of features and a sparse autoencoder (SAE) and a stacked sparse autoencoder (SSAE) is employed for unsupervised learning of features from the mammogram.

2.5 Overall Approach

The overall proposed approach consists of three modules as shown in Fig. 4:

Input Data Generation. The use of entire images as input to the deep network is computationally expensive. Also, downsampling the images may lead to loss of fine details. So, a compact representation can be learned for local neighbors (or

patches) from the image. The size of each patch is limited to 100 × 100 in order to limit the weights and bias to be trained.

Deep Learning Model Representation. The patches are processed by a multilayer convolutional architecture. It consists of three layers in case of deep CNN as shown in Fig. 5. The unsupervised learning network is made of autoencoders. The sparse autoencoder (SAE) consists of four hidden layers, and the stacked sparse autoencoder (SSAE) is composed of three such autoencoders containing four hidden layers each.

Parameter Learning. The parameters of feature representations are obtained by encoding the input using a cascade of trainable transformations. These parameters are learned in a layerwise fashion with or without using the labels. While an individual layer is not deep, the stacked architecture is used. Thus, the individual training of ("shallow") layers results in a deep learning procedure. Figure 6 shows the steps required for transformation of one feature representation to another one.

Finally, a classifier maps the last feature representation into label space. A softmax classifier is used in the proposed system for this purpose. All the extracted patches within the testing image using the sliding window approach are tested by applying the trained hypothesis function. That is, every patch is sent through the trained network for prediction.

2.6 *Risk Scoring*

After the classifier maps the patches into label space, the mammograms can be scored risky or normal based on the density score of the patches. The mammograms

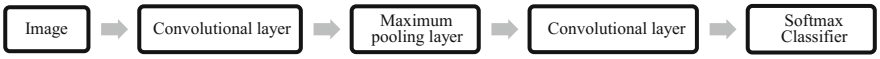


Fig. 5 Multilayer convolutional architecture

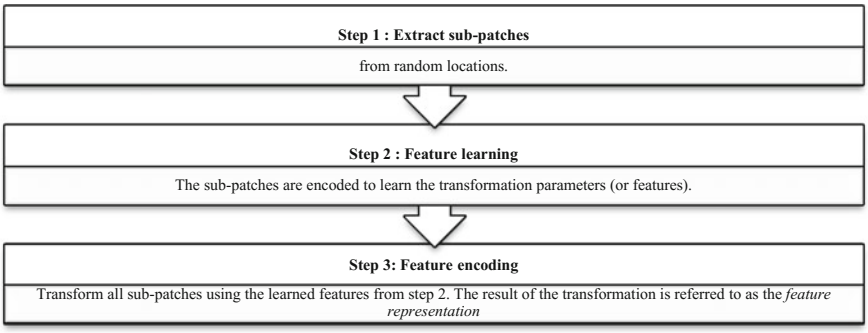


Fig. 6 Feature transformation

are classified as risky, even if at least one of the constituent patches is dense. This technique is similar to that of multiple instance learning.

3 Breast Cancer Imaging Modalities

Different modalities are available for breast cancer screening and detection. Efforts have been made to improve the accuracy of breast cancer diagnosis using different imaging modalities. Figure 7 gives the comparison of different modalities in terms of effective radiation dose to the human body.

Even though the combined PET and CT provide a better view of the suspect cells, the imaging techniques such as CT and PET cause high radiation exposure to patients. In case of using these techniques for screening, the sheer amount of radiation the body is exposed to, while imaging, may itself trigger cancer development. So it is always safer to use imaging techniques such as mammogram and MBI which have comparatively lower effective radiation dose.

In this work, mammogram images are used which is safer in terms of effective radiation dose. Traditional mammograms use X-rays to generate information about the anatomical structure of a lesion.

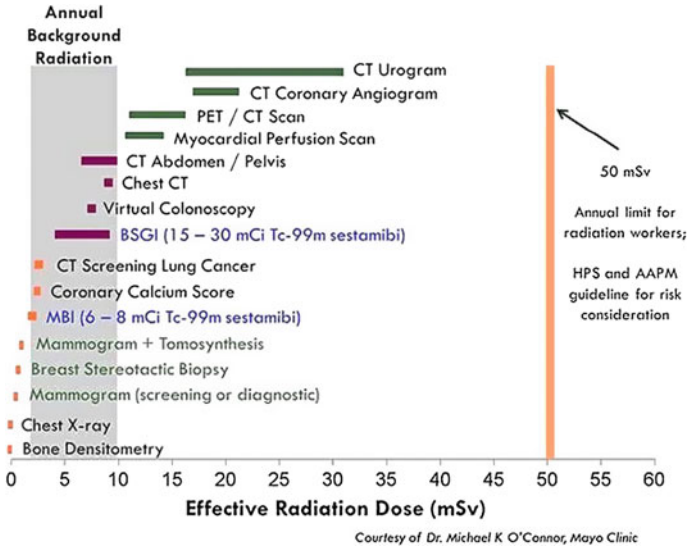


Fig. 7 Effective radiation dose for various medical imaging modalities

4 Deep Learning Techniques

Deep learning techniques can be applied to clustering, classification, and pattern recognition in medicine, biology, robotics, etc. The latest advances in the field have been caused by the invention of deep learning methods. The key component of deep learning is the multilayered hierarchical data representation typically in the form of a neural network with more than two layers. Such methods allow automatically synthesizing data descriptions (features) of a higher level based on the lower ones. A neural network (NN) consists of many interconnected neurons. The number of neurons and the interconnection depends on the application for which it is used. The deep learning techniques can be broadly classified into [10]:

1. *Deep networks for unsupervised or generative learning* (unlabeled data)
2. *Deep networks for supervised learning* (labeled data)
3. *Hybrid deep networks* (combination of 1 and 2).

4.1 Unsupervised Learning

Unsupervised learning is advantageous in cases where there is a huge amount of unlabeled data. By applying unsupervised deep learning techniques to such data, features that are better compared to handcrafted features can be learned.

Autoencoder. An autoencoder is used in unsupervised deep learning techniques. An autoencoder consists of an encoder followed by a decoder as shown in Fig. 8. The encoder part transforms the input x to z using $h^{e}(.)$. The decoder part tries to get back the original input x from z. The decoder outputs $\widehat{x}$ which is the approximate reconstruction of the original input x.

An autoencoder is trained using unlabeled data in an unsupervised fashion, to produce an output which replicates the input. During the training phase, a cost function is minimized. The usually used cost function is cross-entropy as it has lower stagnation periods compared to mean square error.

The encoder part of the autoencoder maps the input vector x, to the vector z using the $h^{e}(\cdot)$ as follows:

$$z^{e} = h^{e}(W^{e}x + b^{e}) \tag{1}$$

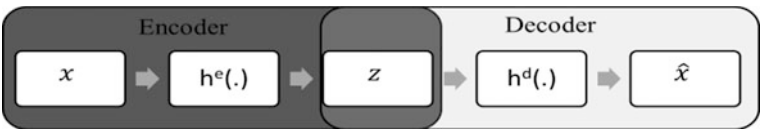


Fig. 8 Autoencoder

where W^{e} is the weight matrix, b^{e} is the bias vector, and $h^{e}(\cdot)$ is the activation function. The activation function may be sigmoid, hyperbolic tangent or rectified linear function. The sigmoid function, $h^{e}(x) = \frac{1}{1+e^{-x}}$, is used in the proposed system. The decoder part of the autoencoder maps the encoded representation z to the approximate reconstruction of the original input vector x using the function $h^{d}(.)$ as follows:

$$\widehat{x} = h^{d}\left(W^{d}x + b^{d}\right) \tag{2}$$

where h^{d} is the transfer function for the decoder, W^{d} is the weight matrix, and b^{d} is the bias vector.

Sparse Autoencoders (SAE). The use of regularizer to the cost function introduces sparsity in an autoencoder [19]. This regularizer is a function of the average output activation value of a neuron. The average output activation measure of a neuron i is defined as follows:

$$\widehat{\rho}_{i} = \frac{1}{n}\sum_{j=1}^{n} z_{i}^{e}\left(x_{j}\right) = \frac{1}{n}\sum_{j=1}^{n} h\left(w_{i}^{eT}x_{j} + b_{i}^{e}\right) \tag{3}$$

where n is the total number of training examples, x_j is the jth training example, w_{i}^{eT} is the ith row of the weight matrix W^{e}, and b_{i}^{e} is the ith entry of the bias vector, b^{e}. A neuron is active, if its output activation value is high. The neuron in the hidden layer fires in response to only a small number of the training examples for a low-output activation value. The neurons can be made to fire to a small number of training values by adding a term to the cost function that constrains the values of $\widehat{\rho}_{i}$. Thus, each neuron specializes by responding to some feature that is only present in a small subset of the training examples.

Sparsity Regularization. A sparse autoencoder can be realized by using a sparsity regularizer with a regular autoencoder. When most of the entries are zero with the exception of only a small number of nonzero entries, this feature representation is called sparse representation. These representations are robust to noise, simple, and cost-efficient. By incorporating population and lifetime sparsity, compact encoding and example-specific features can be obtained.

Sparsity regularizer enforces sparsity on the output from the hidden layer. The sparsity regularization term is made large when the average activation value, $\widehat{\rho}_{i}$, of a neuron i and its desired value, ρ, are not close in value [19]. One such sparsity regularization term is the Kullback–Leibler divergence.

$$\Omega_{\text{sparsity}} = \sum_{i=1}^{D^{e}} \mathrm{KL}(\rho||\widehat{\rho}_{i}) = \sum_{i=1}^{D^{e}} \rho \log\left(\frac{\rho}{\widehat{\rho}_{i}}\right) + (1-\rho)\log\left(\frac{1-\rho}{1-\widehat{\rho}_{i}}\right) \tag{4}$$

Kullback–Leibler divergence is a function which measures the difference between two different distributions. In this case, it takes the value zero when ρ

and $\widehat{\rho}_i$ are equal to each other and becomes larger as they diverge from each other. Minimizing the cost function forces this term to be smaller, hence ρ and $\widehat{\rho}_i$ to be close to each other.

L2 Regularization. When training a sparse autoencoder, it is possible to make the sparsity regularizer small by increasing the values of the weights w^{e} and decreasing the values of z^{e} [19]. Adding a regularization term on the weights to the cost function prevents it from happening. This term is called the L2 regularization term and is defined by:

$$\Omega_{\text{weights}} = \frac{1}{2}\sum_{l}^{L}\sum_{j}^{n}\sum_{i}^{k}\left(w_{ji}^{\mathrm{e}}\right)^2 \tag{5}$$

where L is the number of hidden layers, n is the number of observations (examples), and k is the number of variables in the training data.

Cost Function. The cost function for training a sparse autoencoder is an adjusted mean square error function as follows:

$$E = \frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{K}\left(x_{tkn} - \widehat{x}_{kn}\right)^2 + \lambda * \Omega_{\text{weights}} + \beta * \Omega_{\text{sparsity}} \tag{6}$$

where λ is the coefficient for the L2 regularization term, and β is the coefficient for the sparsity regularization term.

Stacked Sparse Autoencoder (SSAE). A SSAE can be constructed by stacking together multiple layers of basic stacked sparse autoencoders (SAEs) such that the output from the first layer is fed as the input to the next layer. In this work, the SSAE is formed by stacking together three SAEs. Each of these SAEs has four hidden layers. This is followed by a softmax classifier.

4.2 Supervised Learning

Deep networks for supervised learning, which are intended to directly provide discriminating power for pattern classification purposes. It is often done by characterizing the posterior distributions of classes conditioned on the visible data. Target label data are always available in direct or indirect forms for such supervised learning. They are also called discriminative deep networks.

Convolutional Neural Network. A CNN is different from the ordinary backpropagation neural network (BPN) because a BPN works on extracted handcrafted image features, whereas a CNN works directly on an image to extract useful, necessary features for classification. A CNN is usually made up of a number of convolutional layers, pooling layers, fully connected layers, classification layers, etc. When the size of the image given as input to the CNN is $m \times m \times r$, where m is the height and width of the image and r is the number of channels, and

the convolutional layer has k filters of size $n \times n \times q$, k feature maps of size $m - n + 1$ is produced by convolving the image with these filters. Each map is then subsampled typically with mean or max-pooling layers. Subsampling rate usually varies from 2 to 5. After the convolutional layers, there may be any number of fully connected layers.

5 Implementation

The implementation is done using the computer system with MATLAB R2016a having a Windows 10 OS, Intel® Core™ i7-5500 CPU, x64-based processor, 2.40 GHz clock speed, and 8 GB RAM.

5.1 Preprocessing

The procedure for implementing the steps involved in preprocessing of mammogram such as digitization noise removal, artifact suppression, and background removal is discussed below.

Digitization Noise Removal. A 2D median filter of size 3×3 is employed in removing the digitization noises present in the mammogram [4]. In this technique, the center pixel in the 3×3 square covered by the filter mask is replaced by the median of all the 9 pixels in the neighborhood. While computing median in the edges, zero padding is done.

Artifact Suppression and Background Separation. Removal of radio-opaque artifacts present in the mammograms can be done using morphological operations. İn order to convert the mammogram image which is in grayscale to a binary image, the threshold is selected to be $T = 0.0706$. This value of the threshold is found to hold good for most mammograms through manual inspection [4]. The step-by-step procedure involved in artifact suppression and background separation is as follows:

1. First, the objects present in the binary image of the mammogram has to be labeled. The objects include radio-opaque artifacts and the breast profile.
2. The "area" of each object is calculated by counting the number of pixels in the object region.
3. The object with the largest area, which is usually the breast profile region, is selected. Using morphological opening operation, all other objects in the binary image are removed except the breast profile.
4. In order to smoothen the image, isolated pixels have to be removed. This is accomplished by setting a pixel to 1, only when it is surrounded by 5 or more 1 s in a 3×3 neighborhood.
5. The resulting image is first eroded and then dilated using a structuring element of radius 5, which is flat and disk-shaped.

6. Finally, the breast profile is segmented from the background by multiplying the resulting binary image from the previous step with the original grayscale mammogram image. The final grayscale image is devoid of noises and radio-opaque artifacts.

5.2 ROI Segmentation

The pectoral muscle has to be removed from breast profile to have unbiased tumor detection. Seeded region growing (SRG) technique is used for pectoral muscle removal.

Pectoral Muscle Segmentation. The steps involved in the removal of the pectoral muscle region through SRG technique [4] are as follows:

1. *Orientation determination*: As a prerequisite to performing SRG, the breast orientation (left or right) in the mammogram has to be known. The orientation of the breast can be found by comparing the sums of first and last 5 columns in the preprocessed image. If the sum of first five columns is greater than the last five columns, then the breast is right orientated else it is left-orientated.
2. *Seed Point Selection*: Depending on whether the breast is left-oriented or right-oriented, the seed point is selected to perform SRG. The pixel in the 5th row and 5th column is selected as the seed point, if the mammogram is right-oriented. İn case of left-oriented mammogram, the pixel in the 5th row and last 5th column is selected as the seed point.
3. *Seeded Region Growing*: In this technique, the region is grown by comparing the intensity values of the neighboring unallocated pixels. When the difference between these intensity values is less than a threshold value, it is iteratively added to the region. The threshold is selected to be $T = 32$. This value of the threshold is found to hold good for most mammograms for reliable pectoral muscle removal [4]. At the end of SRG, a binary image of the segmented pectoral muscle is obtained.
4. The resulting image is first eroded and then dilated using a structuring element of radius 3, which is flat and disk-shaped.
5. Finally, the binary image obtained from the previous step is multiplied with the preprocessed grayscale image to give the final grayscale mammogram image with the segmented pectoral muscle.

5.3 Cancer Detection Using Deep Learning

The proposed system uses various deep learning techniques to detect breast cancer in mammogram images. The cancer detection using deep learning includes the

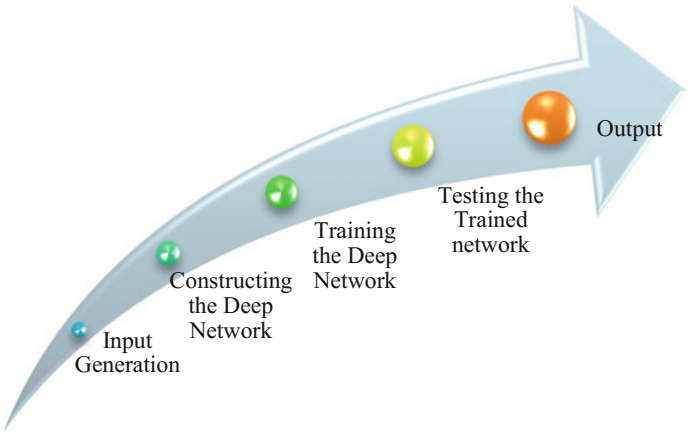


Fig. 9 Steps in cancer detection using deep learning

following implementation steps: input generation, construction of deep network, training the network, and testing the network. The steps in cancer detection are shown in Fig. 9.

Constructing the Deep Network. In this work, three types of deep networks are employed, namely CNN (supervised), SAE, and SSAE (unsupervised).

Convolutional Neural Network. Employing CNNs to classify medical images can be done in three ways. The first method is to build and train the CNN to obtain features. The second method is to use "off-the-shelf CNN features" without retraining the CNN. The third method is to use CNN in fine-tuning the results obtained using other deep learning models.

The first technique is used in building the CNN in this work. The CNN is constructed with 3 layers as shown in Fig. 10. The layers are as follows:

1. Convolutional layer,
2. Maximum pooling layer, and
3. Convolutional layer

The input to the CNN is a 4D matrix containing 100 × 100 image patches. The maximum pooling layer downsamples the convolved image to one-fourth of the original image size. The classification is performed by a softmax classifier.

Sparse Autoencoder. The deep network is constructed by stacking an unsupervised autoencoder and a supervised softmax classifier.

The autoencoder has 4 hidden layers with sparsity parameter $\rho = 0.01$ and weighting term of sparsity regularizer $\lambda = 1$. During the training phase, the autoencoder learns a hypothesis function

$$h : X \rightarrow Y \tag{7}$$

where X is the input image, and Y is the approximate reconstruction of the input image. The encoder part maps the input X to the hidden layers, and the decoder part tries to reconstruct X from the encoded features. The autoencoder applies this hypothesis function h to the input X during the testing phase. Figure 11 shows the deep network that is constructed by stacking autoencoder and softmax classifier.

In the unsupervised part, features are learned using autoencoders. A sparse autoencoder is used in this work. When most of the entries are zero with the exception of only a small number of nonzero entries, this feature representation is called sparse representation. These representations are robust to noise, simple, and cost-efficient. By incorporating population and lifetime sparsity, compact encoding and example-specific features can be obtained.

Stacked Sparse Autoencoder (SSAE). A SSAE can be constructed by stacking together multiple layers of basic stacked sparse autoencoders (SAEs) such that the output from the first layer is fed as the input to the next layer. In this work, the SSAE is formed by stacking together three SAEs. Each of these SAEs has four hidden layers. This is followed by a softmax classifier. The architecture of SSAE is shown in Fig. 12.

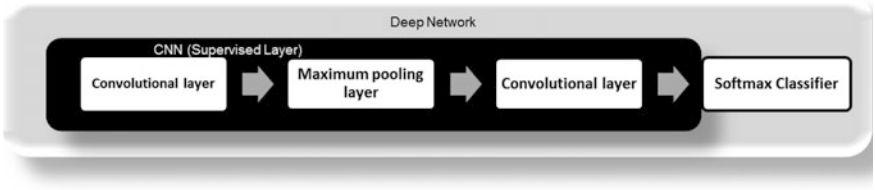


Fig. 10 Convolutional neural network

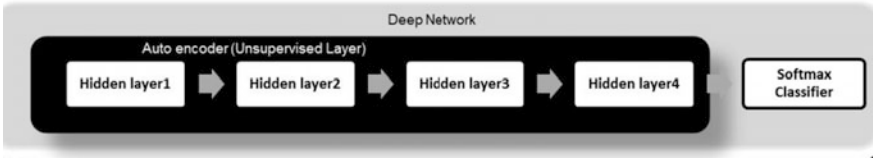


Fig. 11 Sparse autoencoder

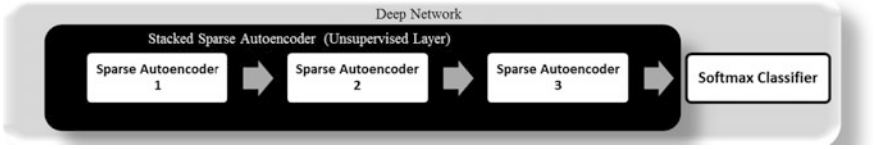


Fig. 12 Stacked sparse autoencoder

Softmax Classifier. Softmax classifier is a supervised layer. Softmax classifier outputs probabilities rather than margins. Probabilities are much easier for us as humans to interpret, so that is a particularly nice quality of softmax classifiers. The regularization term is appended to the loss function and is used to control the weight matrix *W*. By controlling *W*, classification accuracy can be increased. The three class labels for density scoring are background, fatty, and dense. Figure 13 shows the classes in density scoring.

Training the Deep network. Complex classification problems can be solved by using neural networks with multiple hidden layers, where each layer learns different features. Though such networks are attractive for medical image analysis, the training process can be expensive in terms of memory and time.

Greater accuracies of deep learning models can be achieved by using a large amount of training data. However, using such huge amounts of data can slow down the training process. Once a deep learning model is trained, it can be used in real-time applications.

The SAE deep network can be trained by training the autoencoder and softmax classifier separately and then training the deep network after stacking the softmax classifier and SAE. In case of SSAE deep network, each of the autoencoders is trained separately and is again trained after stacking them together. The autoencoder is trained in an unsupervised way, but the softmax classifier and the whole network are trained in a supervised way. Figure 14 depicts the deep network training.

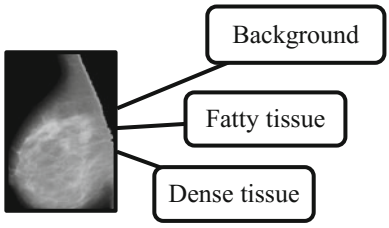


Fig. 13 Softmax classifier classes

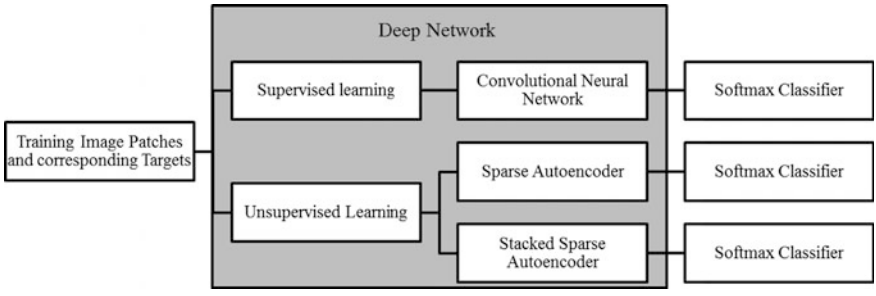


Fig. 14 Training the deep network

Testing the Trained Network. The output of the trained network exactly matches the targets only if the training was done perfectly. If the desired output is not obtained, the network can be trained again. As the network parameters change each and every time the network is trained, the output may differ in a desired way. Another approach is to increase the number of layers of the network because increasing the number of layers increases the flexibility of the network. Or a different training function can be used. Finally, the training data can be increased, as it is likely to improve the performance of the network. Figure 15 depicts the testing of data using the trained deep network.

6 Results and Discussion

There are many different publicly available mammographic databases. The mini-MIAS mammographic database is used in testing the techniques described in this work.

6.1 Database

The database contains 322 digitized films of 1024 × 1024 images. The ground truth markings for all the mammograms are provided by the radiologist, and also, the location of abnormalities is provided.

6.2 Preprocessing

The various preprocessing stages of mammogram are digitization noise removal, artifact suppression, and background removal. The results obtained from each of these stages are discussed in the following sections.

Digitization Noise Removal. A 2D median filtering with 3 × 3 neighborhood window is used to remove the digitization noises present in the input image as

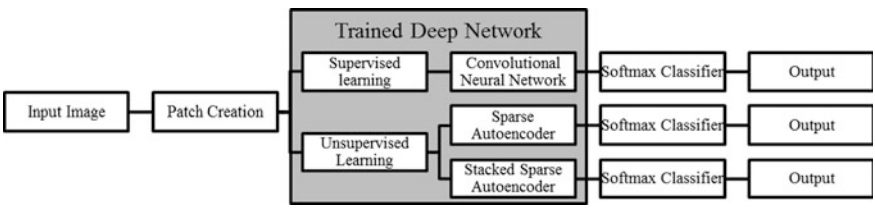


Fig. 15 Deep network testing

shown in Fig. 16a. The resultant image after removal of digitization lines is given in Fig. 16b.

Artifact Suppression and Background Separation. Mammograms with radio-opaque artifacts are shown in Fig. 17a. These artifacts are removed by morphologically opening the binary image. The binary image is shown in Fig. 17b. The resulting binary image after all the morphological operations is done on it as shown in Fig. 17c. The final image is obtained by multiplying Fig. 17a, c, and the final grayscale image is shown in Fig. 17d.

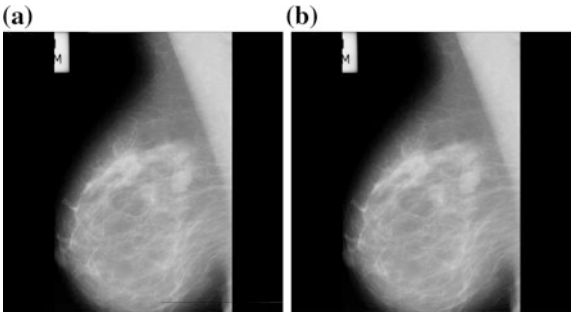


Fig. 16 **a** Original image and **b** Noise-free image

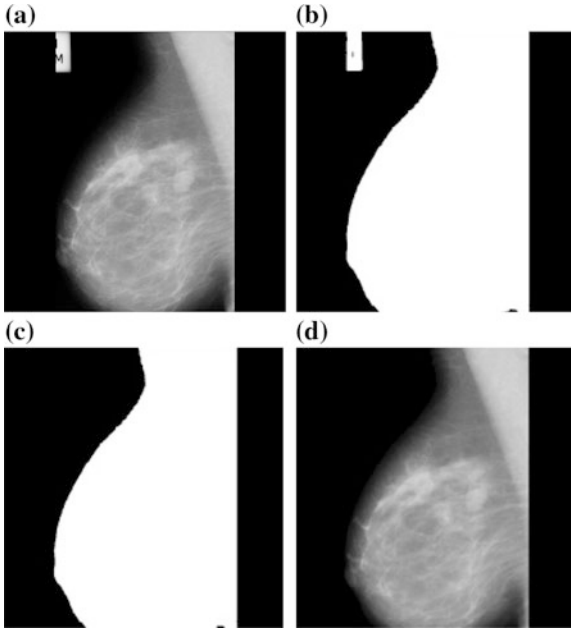


Fig. 17 **a** Noise-free image, **b** binary image, **c** morphologically opened image, and **d** preprocessed image

6.3 ROI Segmentation

The undesired effect of pectoral muscle on the detection of breast cancer warrants it be removed before the detection step. The removal of pectoral muscle is done using SRG technique. The grown region is shown in Fig. 18b. This image is multiplied with the preprocessed grayscale image (Fig. 18a) to get the final grayscale mammogram image with the segmented pectoral muscle shown in Fig. 18c.

6.4 Cancer Detection Using Deep Learning

The cancer detection using deep learning includes the following implementation steps: input generation, construction of deep network, training the network, and testing the network.

Patch Creation. The input image is divided into patches of size 100 × 100 to reduce complexity in computation. The patches extracted from mammogram are shown in Fig. 19. The training patches are selected such a way that there are 10% background patches, 45% fatty tissue patches, and 45% dense tissue patches.

The use of entire images as input to the deep network is computationally expensive. Also, downsampling the images may lead to loss of fine details. So, a compact representation can be learned for local neighbors (or patches) from the image. Thus, images are separated into patches of size 100 × 100 as shown in Fig. 20.

Construction of the Deep Network. The deep networks used in this work are CNN, SAE, and SSAE.

Convolutional Neural Network. The construction of a CNN involves the selection of optimal value for various parameters such as learning rate and number of epochs which influence the accuracy, runtime, etc. The learning rate is set to be

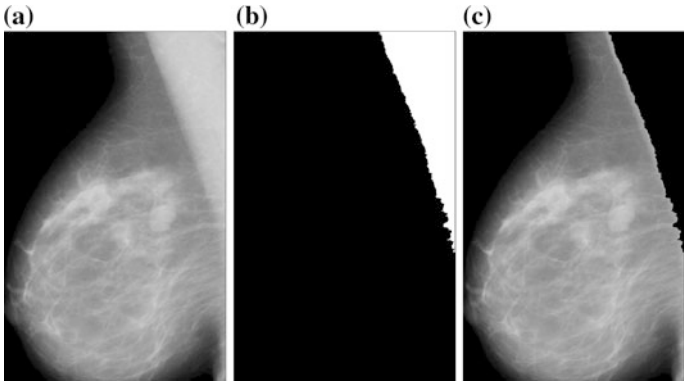


Fig. 18 **a** Preprocessed image, **b** segmented pectoral muscle, and **c** after pectoral muscle removal

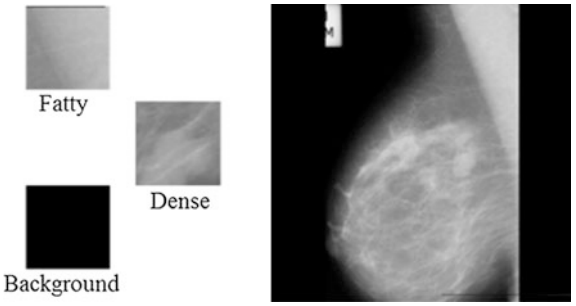


Fig. 19 Patch extraction for mammogram

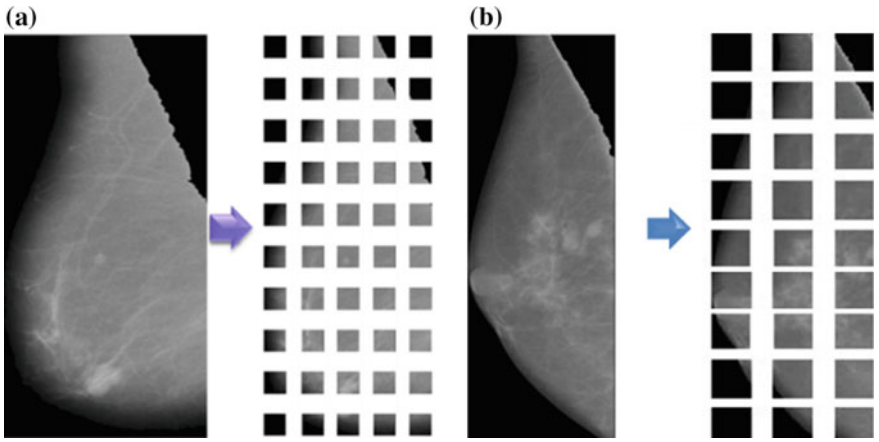


Fig. 20 Mammogram separated into 100 × 100 patches

Fig. 21 Comparison of changing the number of epochs

Number of Epochs	Accuracy	Time
200	77.41	18.65
400	81.85	36.82
600	88.15	54.17
800	93.7	70.99
1000	97	93.86

0.001. Because setting a higher value for learning rate results in suboptimal performance, while lower values increase the training time. Figure 21 shows the effect of varying the number of epochs on the accuracy and runtime. Increasing the number of epochs increases the accuracy, but has longer runtimes.

Sparse Autoencoder. The construction of a sparse autoencoder involves the process of adjusting the network parameters and other settings so that the network

gives optimal results for the current input data. The parameter to be considered while constructing the network includes number of epochs, cost function, decoder function, training algorithm, etc. Figure 22 shows the comparison of various decoder functions in terms of accuracy. By comparing, it is found that linear transfer function gives the best performance in terms of accuracy.

Figure 23 shows the comparison of various cost functions or loss functions in terms of accuracy. By comparing, it is found that minimizing cross-entropy function gives the best performance in terms of accuracy.

Figure 24 shows the comparison of various training algorithms in terms of accuracy and runtime. By comparing, it is found that the resilient backpropagation algorithm gives the best performance in terms of accuracy as well as minimum runtime.

Figure 25 shows the performance comparison in terms of various sizes of training sets in terms of accuracy. By comparing, it is found that accuracy and runtime increase with the increase in the size of the dataset used for training.

6.5 *Performance of the Deep Network*

Performance measurement is done to see whether the output obtained is in line with the target that was intended to be achieved. The performance metrics used for evaluating the performance of the deep network are listed below.

Cross-Entropy Function. The use of cross-entropy function as the minimizing function provides better network performance and short stagnation periods. Thus, the learning time is faster compared to using MSE (mean square error) as minimizing function [20].

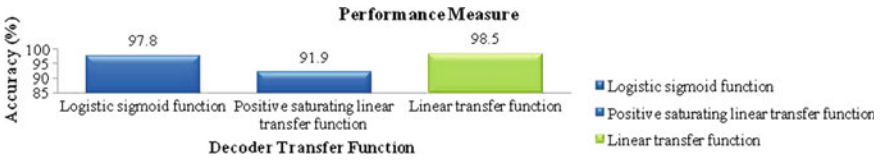


Fig. 22 Comparison of decoder transfer functions

Fig. 23 Comparison of cost functions

Fig. 24 Comparison of training algorithms

Training Algorithm	Accuracy	Time
Levenberg-Marquardt	98.5	29.84
Bayesian Regularization	88	
BFGS Quasi-Newton	98.5	32.62
Resilient Backpropagation	**98.5**	**27.69**
Scaled Conjugate Gradient	98.5	34.73
Conjugate Gradient with Powell/Beale restarts	98.5	35.32
Fletcher-Powell Conjugate Gradient	98.1	
Polak-Ribiere Conjugate Gradient	89.6	
One Step Secant	96.8	
Variable Learning Rate Gradient Descent	90.7	
Gradient Descent with Momentum	93.7	
Gradient Descent	92.2	

Fig. 25 Comparison of size of training sets

$$\text{Cross Entropy} = \frac{1}{m}\sum_{k=1}^{m}[t_k \ln(y_k) + (1 - t_k)\ln(1 - y_k)] \tag{8}$$

where t is the target, and y is the actual output.

Confusion Matrix. The confusion matrix is a two-dimensional table with the output and target classes. This shows whether the system is able to classify the classes correctly or it confuses the classes during classification. Various performance measures such as accuracy, misclassification rate, and specificity can be derived from the confusion matrix.

6.6 Performance Measures

Accuracy and misclassification rate can be calculated from the confusion matrices.

Accuracy. Accuracy is the ratio of correct classifications to the total number of inputs.

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{\text{TOTAL}} \tag{9}$$

where TP—true positive and TN—true negative

Misclassification Rate. Misclassification rate is the ratio of wrong classifications to the total number of inputs.

$$\text{Misclassification Rate} = \frac{(\text{FP} + \text{FN})}{\text{TOTAL}} \tag{10}$$

where FP—false positive and FN—false negative.

Sensitivity or Recall. Sensitivity measures the proportion of correctly identified positives. It is the ratio of true positives to the total correct classifications.

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{11}$$

where TP—true positive, TN—true negative, and FN—false negative.

Specificity. Specificity measures the proportion of correctly identified negatives. It is the ratio of true negatives to the total wrong classifications.

$$\text{Specificity} = \frac{\text{TN}}{(\text{FP} + \text{TN})} \tag{12}$$

where TN—true negative, FP—false positive, and FN—false negative.

Precision. Precision is the ratio of a number of instances correctly classified as positive to all events classified as positive.

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{13}$$

where TP—true positive and FP—false positive.

Negative Predictive Value (NPV). Negative predictive value is the probability of an event classified as negative that is actually negative.

$$\text{Negative Predictive Value} = \frac{\text{TN}}{(\text{FN} + \text{TN})} \tag{14}$$

where TN—true negative and FN—false negative.

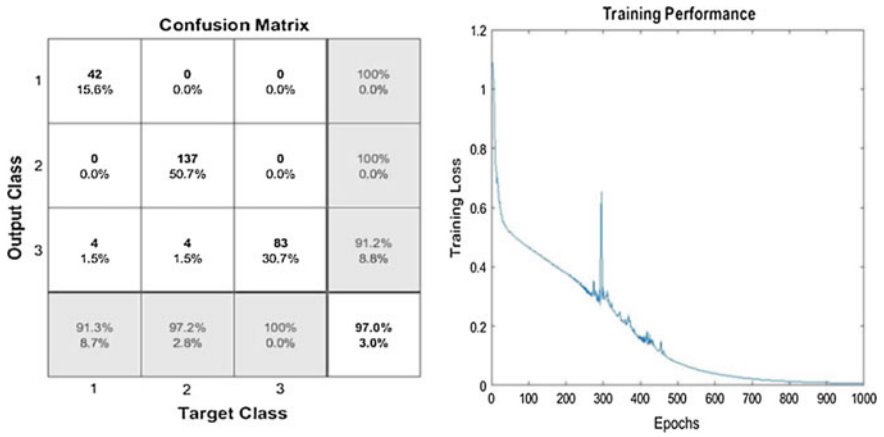


Fig. 26 Confusion plot and training performance in terms of training loss of CNN

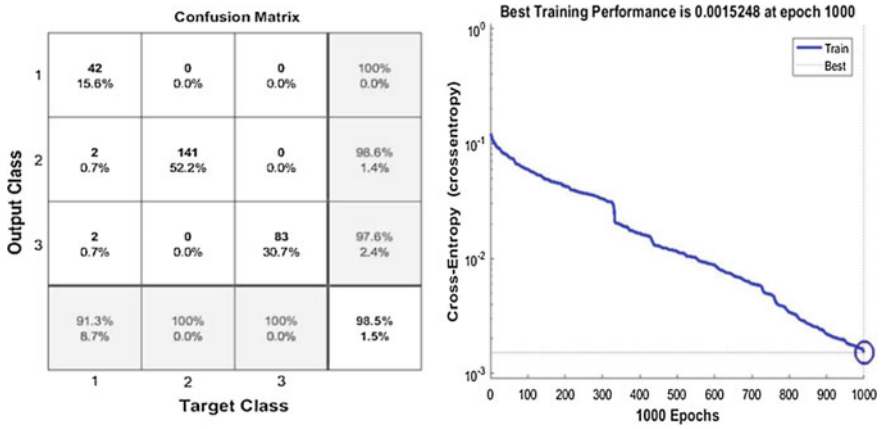


Fig. 27 Confusion plot and training performance of SAE in terms of minimizing the cross-entropy function

Figure 26 shows the confusion plot and training performance in terms of training loss for CNN, and Figs. 27 and 28 show the confusion plots and training performance in terms of cross-entropy for SAE and SSAE, respectively.

The performance of the deep learning techniques such as convolutional neural network, sparse autoencoder, and stacked sparse autoencoder is compared in terms of various performance measures such as accuracy, sensitivity, specificity, precision, and negative predictive value. Figure 29 shows the bar chart comparing the results obtained for various performance measures of these deep networks. From the comparison chart, it can be inferred that the stacked sparse autoencoder performs better than both CNN and SAE in all the cases.

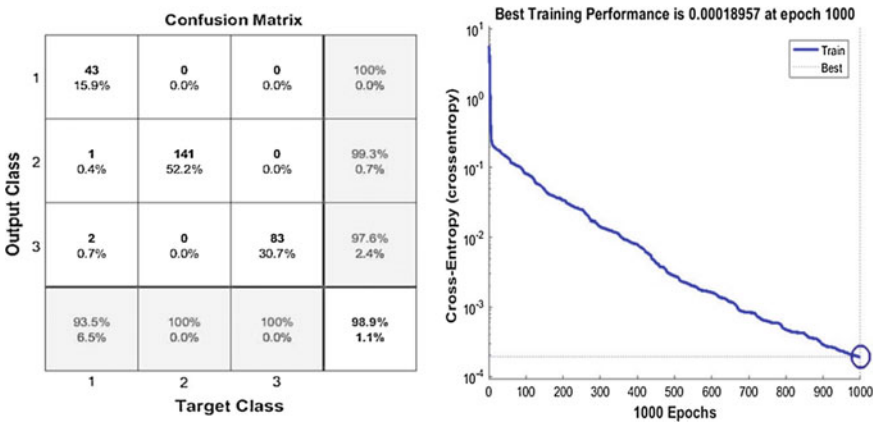
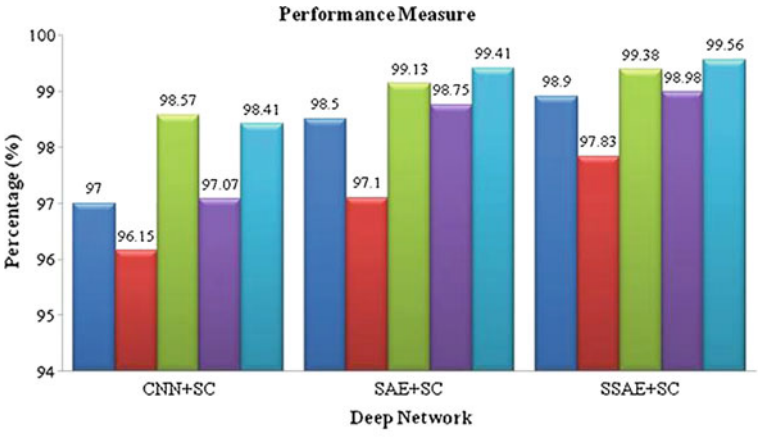
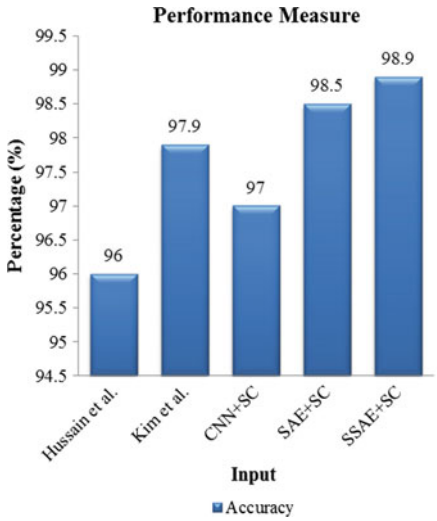


Fig. 28 Confusion plot and training performance of SSAE in terms of minimizing the cross-entropy function



	Accuracy	Sensitivity or Recall	Specificity	Precision	Negative Predictive Value
CNN+SC	97	96.15	98.57	97.07	98.41
SAE+SC	98.5	97.1	99.13	98.75	99.41
SSAE+SC	98.9	97.83	99.38	98.98	99.56

Fig. 29 Comparison of performance of the three deep learning techniques in terms of various performance measures such as accuracy, sensitivity, specificity, precision, and negative predictive value using MIAS database



Abbreviation	Full Form
CNN	Convolutional Neural Network
SAE	Sparse Autoencoder
SSAE	Stacked Sparse Autoencoder
SC	Softmax Classifier

Fig. 30 Comparison of performance of the three deep learning techniques with previous techniques using MIAS database

The performance of the deep learning techniques used in this work (convolutional neural network, sparse autoencoder, and stacked sparse autoencoder) was compared with the performance of the previously proposed techniques which detected breast cancer in mini-MIAS database. The performance was compared in terms of accuracy. Figure 30 shows the bar chart comparing the various techniques. By comparing all these techniques, it can be found that the stacked sparse autoencoder performs better compared to all other techniques.

6.7 Density Scoring

After training the deep network, this trained network is used in predicting the density score of mammogram. The input image is first divided into patches, and the density of each patch is predicted. Figure 31 shows the density scoring results for a mammogram. The mammogram can be classified as abnormal even if any one of the patches extracted from the mammogram is classified as dense. The mammogram is said to be normal only if all the patches constituting the mammogram image are classified either as fatty or background. This technique is similar to that of multiple instance learning (MIL). Figure 31a shows two dense patches which correspond to the dense mass present in the mammogram, and Fig. 31b shows the density scoring of patches obtained from a normal mammogram which shows no dense patches.

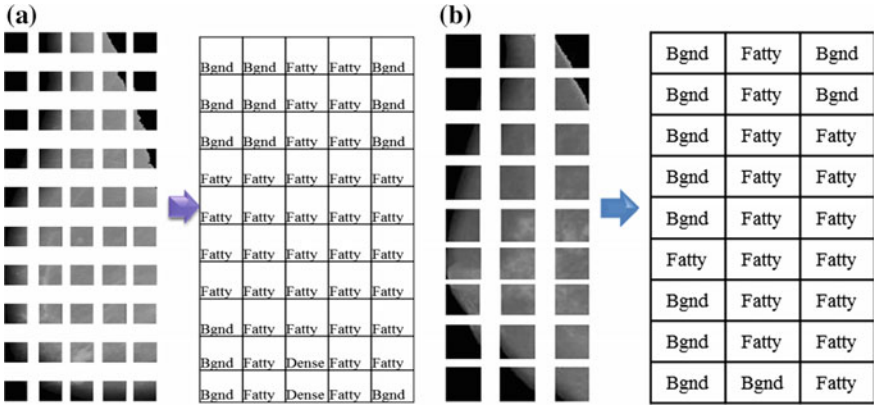


Fig. 31 Density scoring for mammogram

Thus, by comparing both these results, it is clear that only mammograms with no dense patches can be classified as normal. Even if there is an instance of at least one dense patch, the mammogram is classified as dense.

7 Conclusion

In the proposed work, deep learning techniques such as convolutional neural network (CNN), sparse autoencoder (SAE), and stacked sparse autoencoder are used for breast cancer detection using mammographic images. The mammograms from the mini-MIAS database, which is publicly available, are used in this work. The mammograms are first preprocessed to remove digitization noise, radio-opaque artifacts, background, and pectoral muscle, which reduce the effectiveness of the deep network in detecting cancer. The proposed model has achieved an accuracy of up to 97% for CNN (supervised learning), 98.5% for SAE, and 98.9% for SSAE (unsupervised learning) in classifying dense mammogram images.

The use of deep learning techniques in breast cancer detection from mammogram helps to identify smaller masses more accurately to pinpoint their locations, distinguish between cancerous and harmless lesions, and provide quantitative information about these lesions if they are cancerous. This helps in providing earlier treatment for women with breast cancer and spares other women the pain and anxiety of undergoing a biopsy.

References

1. Islam MS, Kaabouch N, Hu WC (2013) A survey of medical imaging techniques used for breast cancer detection. In: IEEE conference on Electro-Information Technology (EIT). IEEE Press, Rapid City, South Dakota, pp 1–5
2. Prasad SN, Houserkova D (2007) The role of various modalities in breast imaging. Biomed Pap Med Fac Univ Palacky Olomouc Czech Repub 151(2):209–218
3. Shyamala K, Girish HC, Murgod S (2014) Risk of tumor cell seeding through biopsy and aspiration cytology. J Int Soc Prev Commun Dent 4(1):5–11
4. Nagi J, Abdul Kareem S, Nagi F, Khaleel Ahmed S (2010) Automated breast profile segmentation for roi detection using digital mammograms. In: IEEE EMBS conference on biomedical engineering & sciences. IEEE Press, Kuala Lumpur, Malaysia, pp 87–92
5. Tan M, Zheng B, Leader JK, Gur D (2016) Association between changes in mammographic image features and risk for near-term breast cancer development. IEEE Trans Med Imaging 35(7):1719–1728
6. Singh AK, Gupta B (2015) A novel approach for breast cancer detection and segmentation in a mammogram. Procedia Comput Sci 54:676–682
7. Pratiwia M, Alexandera, Harefaa J, Nandaa S (2015) Mammograms classification using gray-level co-occurrence matrix and radial basis function neural network. Procedia Comput Sci 59:83–91
8. Kim DH, Choi JY, Ro YM (2012) Region based stellate features for classification of mammographic spiculated lesions in computer-aided detection. In: 19th IEEE international conference on image processing. IEEE Press, Orlando,Florida, pp 2821–2824
9. Hussain M, Khan S, Muhammad G, Bebis G (2012) A comparison of different Gabor features for mass classification in mammography. In: 8th international conference on signal image technology and internet based systems. IEEE Press, Naples, pp 142–148
10. Deng L, Yu D (2014) Deep learning: methods and applications. Now publishers, Boston
11. Schmidhuber J (2015) Deep learning in neural networks: an overview. Neural Networks 61:85–117 Elsevier
12. Langkvist M, Karlsson L, Loutfi A (2014) A review of unsupervised feature learning and deep learning for time-series modelling. Pattern Recogn Lett 42:11–24 Elsevier
13. Guo Y, Liu Y, Oerlemans A, Lao S, Wu S, Lew MS (2016) Deep learning for visual understanding: a review. Neurocomputing 187:27–48 Elsevier
14. Arevalo J, González FA, Ramos-Pollán R, Oliveira JL, Guevara Lopez MA (2015) Convolutional neural networks for mammography mass lesion classification. In: 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE Press, Milan, pp 797–800
15. Sharma K, Preet B (2016) Classification of mammogram images by using CNN classifier. In: International conference on advances in computing, communications and informatics (ICACCI). IEEE Press, Jaipur, pp 2743–2749
16. Baldi P (2012) Autoencoders, unsupervised learning, and deep architectures. In: JMLR: workshop and conference proceedings. pp 37–50
17. Kallenberg M, Petersen K, Nielsen M, Ng AY, Diao P, Igel C, Vachon CM, Holland K, Winkel RR, Karssemeijer N, Lillholm M (2016) Unsupervised deep learning applied to breast density segmentation and mammographic risk scoring. IEEE Trans Med Imaging 35(5): 1322–1331
18. Xu J, Xiang L, Liu Q, Gilmore H, Wu J, Tang J, Madabhushi A (2016) Stacked sparse autoencoder (SSAE) for nuclei detection on breast cancer histopathology images. IEEE Trans Med Imaging 35(1):119–130
19. Olshausen BA, Field DJ (1997) Sparse coding with an overcomplete basis set: a strategy employed by V1. Vision Res 37(23):3311–3325 Elsevier
20. Nasr GE, Badr EA, Joun C (2002) Cross entropy error function in neural networks: forecasting gasoline demand. In: FLAIRS-02 Proceedings. pp. 381–384