Scientific Research Publishing

Journal of Modern Physics, 2024, 15, 1045-1067
https://www.scirp.org/journal/jmp
ISSN Online: 2153-120X
ISSN Print: 2153-1196

Mammogram Classification with HanmanNets Using Hanman Transform Classifier

Jyoti Dabass[1*], Madasu Hanmandlu[2#], Rekha Vig[3], Shantaram Vasikarla[4]

[1]EECE Department, The Northcap University, Sector 23-A, Gurugram, India
[2]CSE Department, MVSR Engineering College, Nadergul, Hyderabad, India
[3]CSE Department, The Northcap University, Sector 23-A, Gurugram, India
[4]CS Department, California State University, Northridge, CA, USA
Email: *mhmandlu@ee.iitd.ernet.in

How to cite this paper: Dabass, J., Hanmandlu, M., Vig, R. and Vasikarla, S. (2024) Mammogram Classification with HanmanNets Using Hanman Transform Classifier. *Journal of Modern Physics*, **15**, 1045-1067.
https://doi.org/10.4236/jmp.2024.157045

Received: March 15, 2024
Accepted: June 25, 2024
Published: June 28, 2024

Abstract

Breast cancer is a deadly disease and radiologists recommend mammography to detect it at the early stages. This paper presents two types of HanmanNets using the information set concept for the derivation of deep information set features from ResNet by modifying its kernel functions to yield Type-1 HanmanNets and then AlexNet, GoogLeNet and VGG-16 by changing their feature maps to yield Type-2 HanmanNets. The two types of HanmanNets exploit the final feature maps of these architectures in the generation of deep information set features from mammograms for their classification using the Hanman Transform Classifier. In this work, the characteristics of the abnormality present in the mammograms are captured using the above network architectures that help derive the features of HanmanNets based on information set concept and their performance is compared via the classification accuracies. The highest accuracy of 100% is achieved for the multi-class classifications on the mini-MIAS database thus surpassing the results in the literature. Validation of the results is done by the expert radiologists to show their clinical relevance.

Keywords

Mammograms, ResNet 18, Hanman Transform Classifier, Abnormality, Diagnosis, VGG-16, AlexNet, GoogleNet, HanmanNets

1. Introduction

The advanced stage of breast cancer is witnessed in the form of a lump or mass

*Corresponding author.
#Formerly with EE Department, IIT Delhi, New Delhi.

in the breast. But to detect at the early stages, screening is important as it helps decide whether to go in for chemotherapy or not with fewer side effects [1]. As not so highly trained radiologists are prone to make wrong decisions, this study aims at an automatic classification of mammograms using deep learning approaches.

The widely used classifiers such as random forest, Bayes, and support vector machine (SVM) have shown good results in the classification of mammograms into benign and malignant. However, these classifiers require the prior annotation of the regions of interest by the expert radiologists, and this limitation is circumvented by using some feature selection methods to improve the classification accuracies. Nowadays, Convolutional Neural Networks (CNNs) have taken over these classifiers by proving their mettle for the twin-tasks of automatic feature extraction and image classification [2] [3] [4]. However, CNNs require a large dataset to avoid either overfitting or poor training results. Moreover, the training of CNNs from scratch is difficult due to the complexity of the type of tumor and the small size of mammograms.

A few applications of deep learning neural networks of which CNNs are the most basic are briefly touched upon for the classification of mammograms. Fine-tuning of residual networks and data augmentation are investigated in [5] on Digital Database for Screening Mammography (DDSM). The residual networks are shown to perform better than the other pretrained networks such as AlexNet, VGG 16, GoogleNet, and Inception V3. The Convolutional Neural Network Improvement for Breast Cancer Classification (CNNI-BCC) in [6] is implemented on the database called mini-Mammographic Image Analysis Society (mini-MIAS) to classify the mammograms into normal, benign, and malignant classes after dividing them into patches followed by the feature-wise data augmentation. A Computer Aided Diagnosis (CAD) system was developed in [7] for the classification of mammograms into benign and malignant categories by finding the Maximally Stable Extremal Regions (MSER) called patches. Fine-tuning of two pretrained model architectures, *i.e.* AlexNet and GoogLeNet is done on the Egyptian National Cancer Institute (NCI), MIAS, DDSM, and INbreast datasets wherein AlexNet has shown better performance. DenseNet 201 is utilized in [8] for the two-class classification of mammograms. For detecting the suspicious lesions, the fine-tuned pretrained CNNs such as AlexNet and PyramidNet are employed in [9].

To learn the discriminative features and to categorize the mammograms from BIRADS (Breast Imaging-Reporting And Data System) into extremely dense, heterogeneously dense, fibro-glandular, and fatty sub-class, an improved CNN framework is presented in [10] by integrating the innovative SE (Squeeze and Excitation)-attention mechanisms. A fine-tuned CNN and a wavelet transform model are developed in [11] for automatically classifying the breast densities into heterogeneously dense and scattered dense. A CAD system in [12] categorizes the mammograms into normal and abnormal classes and the latter into malignant and benign subclasses. It uses Block-based Discrete Wavelet Packet Transform (BDWPT) for feature extraction, Principal Component Analysis (PCA) for

feature reduction, and Weighted Chaotic Salp Swarm Algorithm (WCSSA), and Kernel Extreme Learning Machine (KELM) for feature classification. The Bat Optimized Run-length Network (BORN) is employed in [13] to classify both the MIAS and DDSM datasets.

The parameter optimized KELM in [14] classifies the Haralick texture features extracted using Cross Diagonal Texture Matrix (CDTM) from MIAS and DDSM datasets into normal, benign, and malignant classes. The unknown parameters in KELM are learned with the help of the Grasshopper optimization. The wavelet transform is used in [15] to extract the features that are reduced by Linear Discriminant Analysis (LDA) and PCA. A moth flame optimization coupled with machine learning is applied on the reduced number of features for the two-class classification of the test samples from DDSM.

The mammograms from the mini-MIAS database are classified into abnormal and normal using the transfer learning based on ResNet-18in [16]. The ResNet architecture is also used in [17] for extracting features from a bounding box by enclosing the tumor. It uses the gradient boosted tree for the feature reduction and the two-class (benign and malignant) classification of mammograms. DenseNet II neural network is utilized in [18] to prevent overfitting in the two-class classification. To improve its performance, it makes use of both zero-mean normalization and enhancement. The Inception V3 is employed in [19] to categorize the mammograms into normal, benign, and malignant. In this, two networks in series achieve higher classification accuracies than those of ResNet-18 and DenseNet II.

It is observed from the above survey that the tedious and time-consuming annotations of mammograms produce errors. The lesions' margins are ambiguous and data augmentation aimed at avoiding overfitting increases the time complexity. In the proposed approach, the ROIs (regions of interest) are not needed in the labeling of mammograms thus relieving us from the ambiguous annotations. For this, we make use of the information set concept to modify firstly the kernel functions and feature maps of ResNet architectures, and secondly, the final feature maps of AlexNet, GoogLeNet, and VGG-16 to formulate Type-1 and Type-2 HanmanNets respectively. The deep information set features from these nets are reduced using PCA and then the reduced features are classified by the Hanman Transform Classifier. The workflow of the proposed approach shown in **Figure 1** takes its input from the mini-MIAS database [20] that contains greyscale mammograms each having the size of 1024 * 1024 pixels.

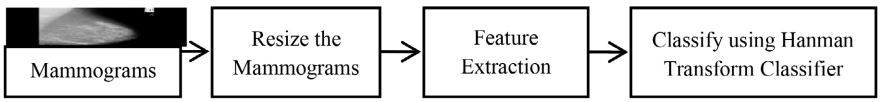

Figure 1. The workflow of the proposed approach.

The objectives of this paper are: 1) To establish a link between the convolution and information set; 2) To modify the architecture of ResNet with the help of the

information set concept; 3) To convert the feature maps of some deep learning network architectures into the deep information set features; 4) To formulate the divergent information set; and 5) To classify the features using the Hanman Transform Classifier.

The rest of the paper is organized as follows. Section 2 describes the CNN architectures such as ResNet, AlexNet, GoogLeNet, and VGG-16. Section 3 provides the formulation of two types of HanmanNets for the extraction of features using the information set concept. Section 4 gives the derivation of divergence information sets needed for the classification. Section 5 presents the development of the Hanman Transform Classifier. Section 6 discusses the results of implementation and conclusions are given in Section 7.

2. Description of Some CNN Architectures

The Convolutional Neural Network (CNN) architectures to be exploited in this work include AlexNet, VGG16, GoogLeNet, and ResNet-18 and the input to these architectures is the resized mammogram.

2.1. The Basic CNN

This consists of convolutional layers where kernels/filters operate to modify the input image/feature map, pooling layers where either selection (max) or aggregation (average) of features is done, fully connected layers where the model is learned, and lastly, a classification layer where a class is labelled by the softmax. As our interest is only in the generation of deep features, we do away with the fully connected layers and softmax. The deep learning architectures used for extracting features are discussed next.

2.2. AlexNet

This has 5 convolutional layers and 3 fully connected layers. The very small weight increments in the gradient based learning needed to learn the unknown parameters of the kernels lead to the problem of vanishing gradients in the saturation region of the sigmoid function and this problem is overcome with the Rectified Linear Unit (ReLU) as the activation function. In this network, the 1st convolutional layer has 96 kernels of size 11 * 11 * 3 and the 2nd has 256 kernels of size 5 * 5 * 48. The 3rd, 4th, and 5th convolutional layers have 384 kernels of size 3 * 3 * 256, 384 kernels of size 3 * 3 * 192, and 256 kernels of size 3 * 3 * 192 respectively followed by the pooling layers. AlexNet has 2 fully connected layers each of size 4096 neurons and a softmax with 1000 class labels. It uses dropout after every fully connected layer to solve the problem of overfitting [21]. As the feature maps from AlexNet are of interest, the last output layer meant for the classification is not considered.

2.3. VGG-16

VGG-16 owing its name to Visual Geometry Group (VGG) has 13 convolutional

(C) layers, 5 max pooling (P) layers, 3 fully connected layers, and its activation function is ReLU [22]. The convolutional and max pooling layers of VGG-16 are sequenced as: 2C, 1P, 2C, 1P, 3C, 1P, 3C, 1P, 3C, 1P where the convolutional layers use kernels of size 3 × 3. The fully connected layers used for classification are removed while extracting features.

2.4. GoogLeNet

Generally, any deep layer network faces the problem of overfitting. GoogLeNet solves this problem by increasing its width. For this, it uses 3 kernels of sizes (5 * 5, 3 * 3, 1 * 1) operating at the same level to capture the variations in different scales. Its architecture has 9 inception modules stacked linearly and 22 layers such that in each layer multiple convolution operations but a single pooling operation are performed. The ends of the inception modules are directly connected to the average pooling layer [23] to get the average of the channels across the feature maps.

Figure 2 shows the naïve inception module where the convolution operations are performed on the inputs with different filters of sizes abbreviated as CONV (5 * 5), CONV (3 * 3), and CONV (1 * 1). The inclusion of CONV (1 * 1) in every layer makes the computation faster by reducing the dimensionality. Also, the max-pooling operation is performed along with convolution operations.

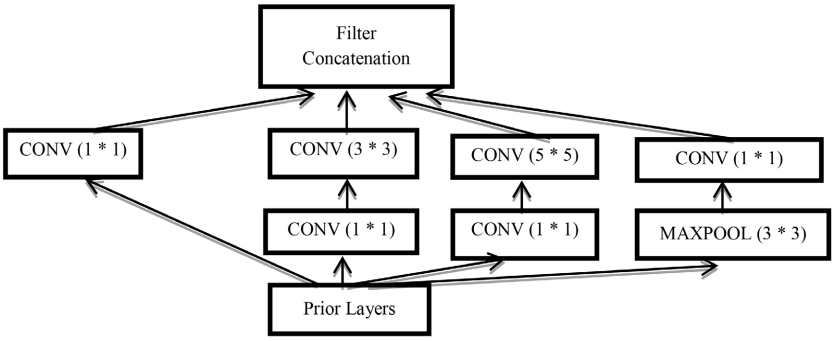

Figure 2. Naïve inception module.

2.5. ResNet-18

The accuracy of the deep learning network increases with an increase in depth as long as overfitting doesn't take place. A deeper network suffers from the problems of degradation, and vanishing gradients during learning. A Residual network that has won the ILSVRC 2015 classification competition solves these problems by having the residual modules and it is found to outperform AlexNet, VGG-16, and GoogLeNet. The residual network is capable of extracting both the low-level and the high-level features [24]. Using transfer learning, one can transfer the features learned from a pretrained CNN to a new classifier assigned with a new task [25] [26] [27]. The output size, N_{output} due to convolution is given by:

$$N_{output} = \frac{N_{input} + 2P - K}{S} + 1 \tag{1}$$

In this, N_{input}, K, and S denote the input size, kernel size and the size of a stride respectively. ResNet is made up of four residual blocks each having 4 convolutional layers along with the two fully connected layers.

Residual Blocks

In place of a regular block, ResNet-18 has a residual block that helps remedy the gradient diminishing problem. Moreover, if any layer in deeper models degrades the performance of its architecture then the residual block allows it to be skipped by regularization [28].

3. Formulation of Two Types of HanmanNets

In this, the Information set concept proposed by Hanmandlu in [29] is used to formulate two types of Deep learning networks christened as HanmanNets. We consider ResNet-18 for the design of Type-1 HanmanNets by modifying its kernel functions as well as the feature maps using the information set concept and next consider any of the three architectures, AlexNet, GoogLeNet, and VGG-16 for the design of Type-2 HanmanNets by modifying its final feature maps. The modified ResNet-18 is symbolized by ResNet (M).

3.1. Establishing a Link between CNN and Information Set Concept

Before formulating the HanmanNets, let us prepare the ground by observing what happens in a convolutional layer of a basic CNN, where we use a kernel to perform a convolution operation. Let the size of a kernel be $k2 = k1 \times k1$ with its weights denoted by w_{ij}, and I^{ij} be the sub-image centred at location (i, j) in the original image I. The convolution is done by centring the kernel mask at every pixel location in an image. Then the ij^{th} convolution operation between I^{ij} and w_{ij} results in the ij^{th} element G_{ij}^{c1} in the reduced image under "valid" size as under:

$$G_{ij}^{c1} = \sum_{i1=1}^{k1} \sum_{j1=1}^{k1} I_{i1,j1}^{ij} w_{i1,j1} \tag{2}$$

By padding zeros both column-wise and row-wise on the image I, we can maintain the "same" size after convolution. The convoluted image $\left\{G_{ij}^{c1}\right\}$ being a kind of membership function matrix is subject to the activation function in the pooling layer by dividing it into blocks of $b1 \times b1$ to consider either the maximum or the average value from each block. This pooling operation leads to the reduced output image, $\left\{G_{ij}^{c1,p1}\right\} = \max\_\text{pool}\left\{G_{ij}^{c1}\right\}$, called the first feature map. In this way, a few convolutional and max-pooling operations go on changing the original image I into a new membership function image $\left\{G_{ij}^{cl,pl}\right\}$ with a considerably reduced size for different combinations of pairs (cl, pl) depending on a particular deep learning architecture chosen.

Information set concept: Consider a set of information source values, say, the pixel intensities, x_{ij} in a sub-image/window of size $n \times n$ centred at location (Ii, Ij) in the original image I, with the corresponding membership function values

denoted by μ_{xij}. The certainty/uncertainty information of x_{ij} is computed using the possibilistic Hanman-Anirban entropy function [29] as follows:

$$H_{li,lj} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{x_{ij}\mathrm{e}^{-\left(a_x\cdot x_{ij}^3+b_x\cdot x_{ij}^2+c_x\cdot x_{ij}+d_x\right)}\right\} \tag{3}$$

If $\mu_{xij} = \mathrm{e}^{-\left(a_x\cdot x_{ij}^3+b_x\cdot x_{ij}^2+c_x\cdot x_{ij}+d_x\right)}$ is assumed to be the Gaussian function, then its parameters are selected as $a_x = 0$, $b_x = \frac{1}{2\sigma_x^2}$, $c_x = \frac{-\bar{x}}{\sigma_x^2}$ and $d_x = \frac{-\bar{x}^2}{2\sigma_x^2}$ where $\bar{x}$ is the mean, σ_x^2 is the variance of pixel intensities covered by the filter mask in the sub-image. With this substitution of parameters, Equation (3) takes the following form:

$$H_{li,lj} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}\mu_{xij} \tag{4}$$

where $\mu_{xij} = \mathrm{e}^{-\frac{\left(x_{ij}-\bar{x}\right)^2}{2\sigma_x^2}}$ is the Gaussian membership function that depends on the mean and variance of the pixel intensities. We can also denote $H_{li,lj}$ by $H(x)$ as it is a function of *x*. The product of the information source/attribute value and its membership function value is called the information value and a set of these values constitutes an information set. The sum of information values in a set gives the certainty information. We will apply this information set concept to the feature maps of any deep learning network to derive the deep information set features later.

Let us consider the adaptive Hanman-Anirban entropy function in which the parameters are assumed to be variables, *i.e.* they are functions of x_{ij}. This is expressed as:

$$Ha_{li,lj} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}\mathrm{e}^{-\left(a_x(ij)\cdot x_{ij}^3+b_x(ij)\cdot x_{ij}^2+c_x(ij)\cdot x_{ij}+d_x(ij)\right)} \tag{5}$$

Looking at the adaptive exponential gain function in Equation (5), $\mathrm{e}^{-\left(a_x(ij)\cdot x_{ij}^3+b_x(ij)\cdot x_{ij}^2+c_x(ij)\cdot x_{ij}+d_x(ij)\right)}$, we find that there are two parts: the first part is a function of *x* and the second part is independent of *x*. To make the dependent part zero, we substitute $a_x(ij) = b_x(ij) = c_x(ij) = 0$ and take the independent part as $d_x(ij) = 1 - \sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}w_{ij}$, then the gain function becomes $\mathrm{e}^{-1-\sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}w_{ij}} = \sum_{i=1}^{n}\sum_{j=1}^{n}x_{ij}w_{ij}$ on taking the first two terms of the exponential function and this is similar to the r.h.s. of Equation (2) thus representing G_{ij}^{c1}. But, the element G_{ij}^{c1} is obtained using the kernel function on the sub-image of size $k1 \times k1$ in the c1-convolutional layer. That is, applying the exponential gain function is equivalent to performing the convolution operation on the sub-image by a kernel function. That is, with an appropriate substitution for the variables in (5) we get whatever a kernel function bestows us on convolving it with the sub-image. This proves that the convolution of the sub-image with a kernel yields an approximate form of the information set consisting of a set of information values $\{x_{ij}w_{ij}\}$.

3.2. Type-1 HanmanNets from ResNet

Let us examine the regular block-based CNN based architecture shown in **Figure 3(a)** and how it differs from a residual block in ResNet-18 shown in **Figure 3(b)** in the sense that the latter has a feedback loop. This prompts us to use three properties of an information set [30] [31], given as under:

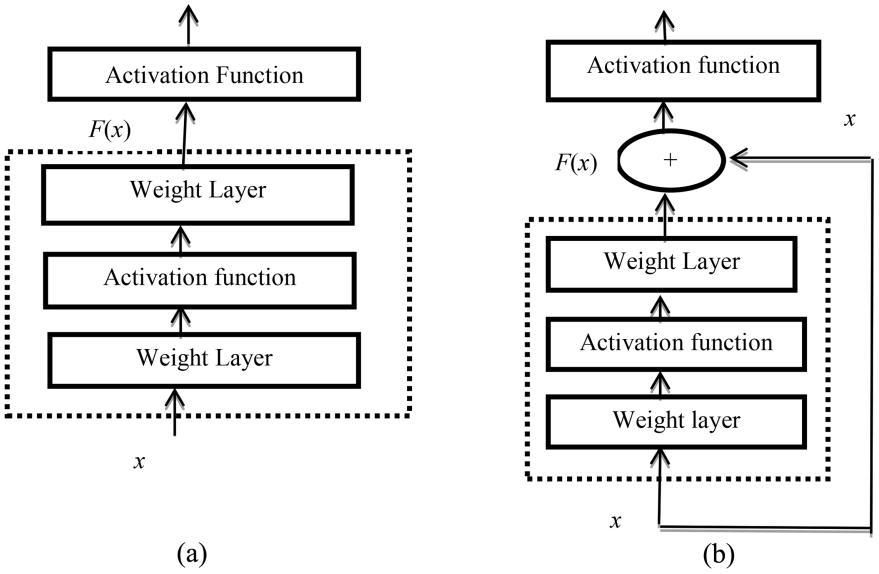

Figure 3. Blocks (a) Regular block on the left, (b) Residual block on the right.

1) The membership function value can be changed by adding an attribute value or a random number to it.

2) Each certainty information value $H_{ij} = \left(x_{ij}\mu_{xij}\right)$ considered as the unit of information tells us how much the information source/attribute value, x_{ij} is associated with a concept called the degree of association, μ_{xij} and this unit of information can be changed by applying a function such as sigmoid and log.

3) New information can be made to depend on the old information values.

ResNet-18 has "skip connection" facility that permits skipping layer/layers and this helps to solve the gradient vanishing problem. As we have established a link between the convolution where a kernel mask acts on a sub-image/window and an information set where a membership function acts on the pixel intensities in a sub-image/window, the properties of information set can be utilized in ResNet-18 to yield the modified ResNet-18 now called ResNet (M).

As per the first property, the membership function value can be changed as $\mu'_{xij} = \mu_{xij} + x_{ij}$. This turns out to be what have in **Figure 3(b)**, *i.e.* $F'\left(x_{ij}\right) = F\left(x_{ij}\right) + x_{ij}$. The operations of deep learning such as convolution, pooling, skipping etc. performed on the pixel intensities x_{ij} of input image are represented by $F\left(x_{ij}\right)$ to which we are adding x_{ij}. If $F\left(x_{ij}\right)$ is an element of the feature map then $F'\left(x_{ij}\right)$ is the modified element after the feedback. Using the information set concept that deals with the operations on the information values, we can find the modified certainty information value using Equation (4)

as:

$$H_{ij} = H\left(x_{ij}\right) = x_{ij} \cdot \mu'_{xij} = x_{ij} \cdot F'\left(x_{ij}\right) = x_{ij} \cdot F\left(x_{ij}\right) + x_{ij}^2 \tag{6}$$

where x_{ij} is the pixel intensity value of the reduced original image and $F\left(x_{ij}\right)$ is the corresponding membership function value from the feature map of ResNet (M). architecture selected. Thus, H_{ij} is the feature of Type-1 HanmanNet denoted by HanmanNet-1R.

3.3. Sigmoid HanmanNet

As per Property-ii of the information set, we can also modify the certainty information value, H_{ij} by applying the sigmoid function to yield the sigmoid information value as follows:

$$SG_{ij} = 1 \Big/ \left(1 + \exp\left(-H_{ij}\right)\right) \tag{7}$$

Thus, SG_{ij}, called the sigmoid HanmanNet, is another feature of Type-1 HanmanNet denoted by HanmanNet-1S. We will now attempt at the formulation of Type-2 HanmanNets from the CNN architectures that we have taken up for study in this work.

3.4. Type-2 HanmanNets

In ResNet (M), a kernel function is changed by adding a random value due to skip connection. In AlexNet, GoogLeNet, and VGG-16, we will only modify the final feature map/output image resulting from the application of kernels in every convolutional layer followed by their selection in every pooling layer. The size of the feature map is considerably reduced and the final size depends on the number of convolutional layers including the sizes of the filters chosen and of the max-pooling layers involved in the architecture. To be compatible with the feature map, the size of the original image is reduced to that of the feature map. Then Type-2 HanmanNets features are obtained as $\left\{x_{ij}F\left(x_{ij}\right)\right\}$ on considering the corresponding pixel intensity values in both the reduced original image x_{ij} and the feature map of the chosen CNN architecture. The choice of the final feature maps of AlexNet, GoogLeNet, and VGG-16 yields us the features of Type-2 HanmanNets-2A, HanmanNets-2G and HanmanNets-2V respectively on applying the information set concept. The advantage of Type-1 HanmanNet is that we can skip layers and tap the features from any layer. But in Type-2 HanmanNets, the features are tapped from only the final feature map to avoid the experimentation.

So far, we have dealt with the extraction of features from the input mammograms of patients using the two types of HanmanNets. But in the context of classification into different classes by the proposed Hanman Transform Classifier, we need to have the features from both the training and test sets. So, we shall prepare the ground to utilize these two sets of features in the classification. Let $\left\{x_{ij}\right\}$ and $\left\{y_{ij}\right\}$ be the reduced original images to the size of the feature map in the training and test sets respectively.

Type-1 HanmanNet: In ResNet (M), the feature maps of the training and test images, $F'(x_{ij})$ and $F'(y_{ij})$ are modified for each dropout. At the final dropout, the training and test information values are obtained as $x_{ij} \cdot F'(x_{ij}) = x_{ij} \cdot F(x_{ij}) + x_{ij}^2$ and $y_{ij} \cdot \mu'_{yij} = y_{ij} \cdot F'(y_{ij}) = y_{ij} \cdot F(y_{ij}) + y_{ij}^2$ respectively. So, we compute the absolute difference between their respective information values called the divergence because of using two agents, μ'_{xij} and μ'_{yij} which are the pixel intensities in the feature maps of ResNet (M).

$$e_{1R}(ij) = \left| x_{ij} \cdot \mu'_{xij} - y_{ij} \cdot \mu'_{yij} \right| \tag{8}$$

The above subscript 1R refers to Type-1 divergent ResNet (M) information value. It may be noted that $\{e_{1R}(ij)\}$ is the divergent information set. If we use the sigmoid information values, then we have Type-1 divergent sigmoid information value denoted by the subscript 1S, as:

$$e_{1S}(ij) = \left| S(x_{ij} \cdot \mu'_{xij}) - S(y_{ij} \cdot \mu'_{yij}) \right| \tag{9}$$

Type-2 HanmanNets: In AlexNet, GoogLeNet, and VGG-16, the final feature maps of training and test samples contain at ij^{th} location: $F(x_{ij}) = \mu_{xij}$ and $F(y_{ij}) = \mu_{yij}$ respectively. As in Type-1 HanmanNets, we also compute the absolute differences between their respective information values leading to Type-2 divergent information values:

$$e_2(ij) = \left| x_{ij}\mu_{xij} - y_{ij}\mu_{yij} \right| \tag{10}$$

With another subscript included in $\{e_2(ij)\}$ like $\{e_{2A}(ij)\}$, $\{e_{2G}(ij)\}$ and $\{e_{2V}(ij)\}$, it refers to Type-2 AlexNet, GoogLeNet and VGG-16 divergent information sets denoted by Type-2 HanmanNet-2A, Type-2 HanmanNet-2G and Type-2 HanmanNet-2V respectively.

Next, we present another approach for dealing with the deep information set features of the training and test sets during classification.

4. Derivation of Conditional Divergent Information Set

To achieve this objective, the need arises here to invoke the possibilistic Hanman-Anirban cross entropy function, expressed as:

$$HC(x,y) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ x_{ij} \mathrm{e}^{-\left(a_y \cdot x_{ij}^3 + b_y \cdot x_{ij}^2 + c_y \cdot x_{ij} + d_y\right)} \right\} \tag{11}$$

If we choose $a_y = 0$, $b_y = \frac{1}{2\sigma_y^2}$, $c_y = \frac{-\overline{y}}{\sigma_y^2}$ and $d_y = \frac{-\overline{y}^2}{2\sigma_y^2}$ in Equation (11) where $\overline{y}$ is the mean and σ_y^2 is the variance of pixel intensities in a sub-image, we get:

$$HC(x,y) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij}\mu_{yij} \tag{12}$$

If $HC(x,y)$ is a function of x alone then it becomes $H(x)$ as defined in Equation (4). Both Equations (4) and (12) are derived based on the information set concept. Our objective is to establish a link between the information set con-

cept and deep learning network architecture. In the context of deep learning architecture, the elements of feature maps are treated as membership function values, viz., μ_{xij} (or μ'_{xij}) and μ_{yij} (or μ'_{yij}) because of modification of the input image through kernel functions. Now by subtracting Equation (4) from (12), we get what we call the conditional divergent entropy function, expressed as:

$$\begin{aligned} D_c(x \mid y) &= H(x) - HC(x,y) \\ &= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\mu_{xij} - \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\mu_{yij} \\ &= \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\left(\mu_{xij} - \mu_{yij}\right) \end{aligned} \tag{13}$$

The divergence of information is the result of looking at an object (here, an attribute value, x_{ij}) by two agents (here, the two membership functions, μ_{xij} and μ_{yij}), differently.

The Conditional Divergent Information set: The conditional divergent information set is obtained by considering the absolute values of $x_{ij}\left(\mu_{xij} - \mu_{yij}\right)$ from Equation (13) as:

$$D(x \mid y) = \left\{\left|x_{ij}\left(\mu_{xij} - \mu_{yij}\right)\right|\right\} \tag{14}$$

The higher form of Hanman cross-entropy function: This is derived from the adaptive Hanman-Anirban cross-entropy function that results from Equation (11) if its parameters are variables, as given by:

$$HC_a(x,y) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{x_{ij}\mathrm{e}^{-\left(a_y(ij)\cdot y_{ij}^3 + b_y(ij)\cdot y_{ij}^2 + c_y(ij)\cdot y_{ij} + d_y(ij)\right)}\right\} \tag{15}$$

Substituting $a_y(ij) = b_y(ij) = d_y(ij) = 0$ and $c_y(ij) = \mu_{yij}$ in (15), we get

$$HC_a(x,y) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\mathrm{e}^{-\mu_{yij}\cdot y_{ij}} \tag{16}$$

This is termed as the Hanman cross transform and to derive the Hanman transform we need to replace *y* with *x* in the exponential gain function of Equation (15). This replacement leads to the expression given by:

$$H_a(x) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{x_{ij}\mathrm{e}^{-\left(a_x(ij)\cdot y_{ij}^3 + b_x(ij)\cdot y_{ij}^2 + c_x(ij)\cdot y_{ij} + d_x(ij)\right)}\right\} \tag{17}$$

Substituting $a_x(ij) = b_x(ij) = d_x(ij) = 0$ and $c_x(ij) = \mu_{xij}$ in Equation (17) gives us the Hanman transform:

$$H_a(x) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\mathrm{e}^{-\mu_{xij}\cdot x_{ij}} \tag{18}$$

The possibilistic conditional divergent Hanman transform is defined as:

$$DH_a(x \mid y) = H_a(x) - H_a(x,y) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij}\left(\mathrm{e}^{-\mu_{xij}\cdot x_{ij}} - \mathrm{e}^{-\mu_{yij}\cdot y_{ij}}\right) \tag{19}$$

where $\{x_{ij}\}$ and $\{y_{ij}\}$ are the original reduced images. The corresponding conditional divergent Hanman transforms of Type-1 and Type-2 deep information set values at the pixel level are denoted by:

$$DH_1(ij) = \left| x_{ij} \left(e^{-x_{ij} \cdot \mu'_{xij}} - e^{-y_{ij} \cdot \mu'_{yij}} \right) \right| \tag{20}$$

$$DH_2(ij) = \left| x_{ij} \left(e^{-x_{ij} \mu_{xij}} - e^{-y_{ij} \cdot \mu_{yij}} \right) \right| \tag{21}$$

As we know that the sigmoid function of information value is more effective than the exponential function of the information value, we replace the exponential gain function with the sigmoid function in Equation (19) to get the conditional divergent sigmoid transforms of Type-1 and Type-2 deep information set values at the pixel level, denoted by:

$$DS_1(ij) = \left| x_{ij} \cdot \left\{ SG\left(x_{ij} \cdot \mu'_{xij} \right) - SG\left(y_{ij} \cdot \mu'_{yij} \right) \right\} \right| \tag{22}$$

$$DS_2(ij) = \left| x_{ij} \cdot \left\{ SG\left(x_{ij} \mu_{xij} \right) - SG\left(y_{ij} \mu_{yij} \right) \right\} \right| \tag{23}$$

It may be noted that $DH_1(ij)$ and $DS_1(ij)$ make use of Type-1 HanmanNet-1R, 1S features whereas $DH_2(ij)$ makes use of Type-2 HanmanNets-2A, 2G, 2V features. In this work, we have not implemented $DS_2(ij)$ as it involves more computations than $DS_1(ij)$.

The use of Divergent Information set: Note that each input mammogram gives rise to several feature maps due to the use of kernels at the convolutional layers but their sizes get reduced at the max pooling layers. As a result, the finally selected feature maps have considerably reduced sizes. We aggregate all the feature maps into one and then the size of the input image is reduced to the size of the aggregated feature map. Now, the pixel intensities of the reduced input image serve as the information source/attribute values whereas those of the aggregated feature serve as the membership function values with their products becoming the information values as per the information set concept. Using these information values we form the feature vectors. During classification, we need a number of training feature vectors (*i.e.* U) each of length, N for z^{th} class, but only one test feature vector, so we consider the reduced input images and the aggregated feature maps in vector form to compute the feature vectors. Let X^z_{UJ} be the reduced training input vector and μ'_{XUJ} (ResNet (M)) or μ_{XUJ} (CNN) be the training membership function vector. Let Y_J be the reduced test input vector and μ'_{YJ} or μ_{YJ} be the test membership function vector. The training feature vector is obtained as $X^z_{UJ}\mu'_{XUJ}$ or $X^z_{UJ}\mu_{XUJ}$ whereas the test feature vector is $Y_J\mu'_{YJ}$ or $Y_J\mu_{YJ}$. The absolute difference $\left| X^z_{UJ}\mu_{XUJ} - Y_J\mu_{YJ} \right|$ is used as the error vector in the Hanman classifier [32]. In this work, we compute the transformed feature values of training set denoted by $g^z_{UJ} = e^{-X^z_{UJ}\mu'_{XUJ}}$ or $e^{-X^z_{UJ}\mu_{XUJ}}$ and test set by $t_J = e^{-Y_J\mu'_{YJ}}$ or $e^{Y_J\mu_{YJ}}$.

5. An Algorithm for the Hanman Transform Classifier

The Hanman Transform Classifier as described in [32] is well suited to small datasets. In this classifier, the absolute error vectors between the training feature vectors of a class and a test feature vector are computed, and then T-norm is applied on the two error vectors at a time to obtain all possible T-normed error

vectors for each class. Out of all the T-normed error vectors of a class, the one with the least Hanman transform value is selected as representing the class since this is on the boundary between its class and the neighbouring class and all other vectors are all within the class. This vector is similar to a support vector in the support vector machine and the Hanman transform operating on the T-normed error vectors is taken as the criterion function. Hence, this Hanman transform differs from the Hanman transform that is used to transform the information values above. Next, the infimum of the Hanman transform values of the selected T-normed error vectors of all classes gives the identity of the test class [32]. However, we envisage here the use of the transformed conditional divergent deep information values in place of the divergent information values used in [32].

Algorithm for the Hanman Transform Classifier

Setp 1: Compute the divergent vector using the conditional divergent deep information values from the training feature vectors of z^{th} class and a test feature vector as:

$$e_{UJ}^{z} = \left| X_{UJ}^{z} \left(g_{UJ}^{z} - t_{J} \right) \right| \tag{24}$$

Here $U = 1, 2, \cdots, Nf$ (the number of feature vectors) and $J = 1, 2, \cdots, N$ (the size of a feature vector).

Setp 2: Compute T-normed divergent vector for z^{th} class from a (U, M) pair of error vectors using:

$$E_{UM}^{z}(J) = T\left(e_{UJ}^{z}, e_{MJ}^{z} \right); U \neq M \tag{25}$$

where $(x, y) = \log_{P}\left(1 + \dfrac{\left(P^{x} - 1\right)\left(P^{y} - 1\right)}{P - 1} \right)$ with P generally taken as $P = 2$, spans the space of Frank T-norms.

Setp 3: Compute the exponential membership function of T-normed divergent vector as,

$$\mu_{UM}^{z}(J) = \mathrm{e}^{-E_{UM}^{z}(J)} \tag{26}$$

Setp 4: Estimate $T_{UM}(z)$ using the Hanman transform for each class using,

$$T_{UM}(z) = \sum_{q=1}^{N} E_{UM}^{z}(q) \mathrm{e}^{-E_{UM}^{z}(q)\mu_{UM}^{z}(q)} \tag{27}$$

where $U \neq M$ and $U, M = 1, 2, \cdots, Nf$, and $z = 1, 2, \cdots, c$ (*i.e.* the number of classes).

Setp 5: Repeat Steps 1 - 4 for each class.

Setp 6: Compute the minimum of $T_{UM}(z)$ for each z and denote it by $T(z)$. The selected T-normed divergent vectors from this criterion are termed as the support vectors.

Setp 7: Calculate I = Infimum of $\{T(z)\}$ then label the test feature vector of z^{th} class. The selected support vector corresponding to I gives the identity of the class.

6. Results and Discussions

6.1. Quantitative Measures

For evaluating the efficiency of the proposed results, we have used sensitivity, precision, specificity, F-score, and accuracy. The evaluation indexes are denoted by TN (true negative), FP (false positive), TP (true positive), and FN (false negative). Specificity is the ratio of correctly detected TN negative labels to the total number of (TP + FN) actual negative labels whereas sensitivity or recall is the ratio of correctly detected TP positive labels to the total number of (TP + FN) actual positive labels. Precision quantifies the correctly detected TP positives to all positives including true positives and false positives. The harmonic mean of recall and precision gives a score called F-score. Accuracy is the ratio of the correct labels (TP + TN) to the set of all labels (TP + FN + TN + FP). The following equations are used for computing these measures.

$$\text{Precision} = \frac{\text{TP}}{\text{FP} + \text{TP}} \tag{32}$$

$$\text{Sensitivity}\left(\text{Recall}\right) = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{33}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \tag{34}$$

$$\text{F\_score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \tag{35}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TN} + \text{FN} + \text{FP} + \text{TP}} \tag{36}$$

6.2. Results of Implementation

As mentioned above we have used a digital mammogram from the mini-Mammographic Image Analysis Society (MIAS) for ascertaining the performance of the proposed approach. This dataset consists of 322 mammograms of 161 women. Each mammogram is of size 1024 * 1024 pixels with an 8-bit grey level resolution having ground truth provided by the expert radiologists. It contains 208 normal, 53 malignant, and 61 benign mammograms. The ground truth of the dataset contains the characteristics of background tissues (fatty, glandular or dense), location of the abnormality, the severity of each abnormality (benign, malignant), the radius of a circle enclosing the abnormal regions, and the type of abnormality present (calcification, normal, asymmetry, ill-defined masses, spiculated masses, well defined/circumscribed masses, and architectural distortion) [20]. Some sample mammograms of these classes are shown in **Figure 4**.

In the two-class classification of mammograms, two stages are involved. In the first stage, mammograms are classified into normal and abnormal, and in the second stage, abnormal mammograms are classified into benign and malignant. Out of 322 mammograms, 276 mammograms are set apart for classification into normal and abnormal and out of 276, 184 mammograms are used for training and the rest for testing. Next, 114 abnormal mammograms need to be split up

into benign and malignant classes. In this case, 76 mammograms are kept for training and the rest for testing. Table 1 gives the classification accuracies achieved by the Hanman Transform Classifier on features from the two types of HanmanNets. For simplicity of notation, we had denoted above the modified ResNet-18 by ResNet (M), Type-1 HanmanNet (ResNet) by HanmanNet-1R, Type-1 Sigmoid HanmanNet by HanmanNet-1S, Type-2 HanmanNet (AlexNet) by HanmanNet-2A, Type-2 HanmanNet (GoogLeNet) by HanmanNet-2G and Type-2 HanmanNet (VGG-16) by HanmanNet-2V.

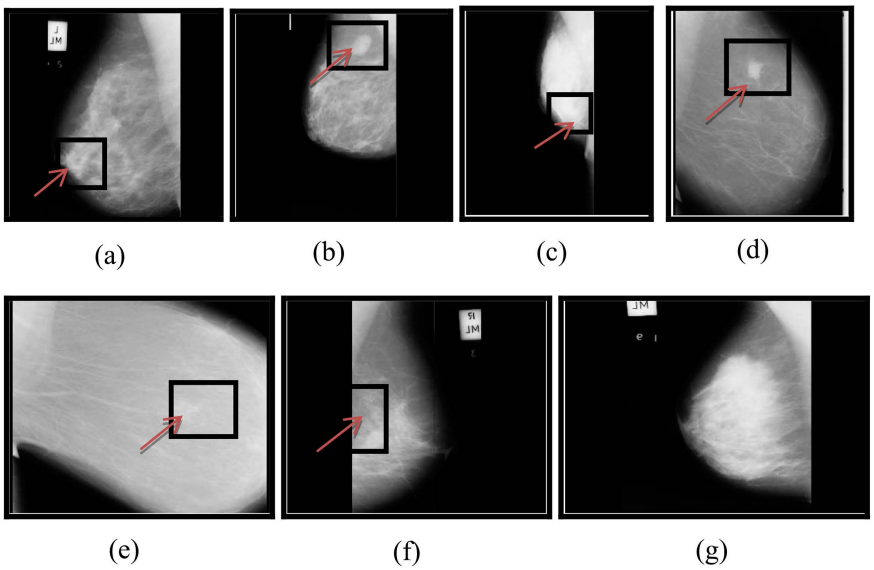

Figure 4. Sample of mammograms used for classification (a) Mdb219 with benign microcalcification on fatty glandular background tissue shown by an arrow, (b) Mdb015 with benign circumscribed/well-defined mass on fatty glandular background tissue, (c) Mdb179 with the spiculated malignant mass on dense background tissue, (d) Mdb134 with the ill-defined malignant mass on fatty background tissue, (e) Mdb152 with benign architectural distortion on fatty background tissue, (f) Mdb104 with malignant asymmetry on dense background tissue, (g) Mdb003 normal mammogram on dense background tissue.

Table 1. Classification accuracies achieved by the Hanman Transform Classifier (in percentage) for the multi-class classifications using the features of Type-1 and Type-2 HanmanNets.

Architecture	Normal/ Abnormal	Benign/ Malignant	F/G/D	N/B/M	Six class	Seven Class
ResNet (M)	100	97.36	100	97.72	100	92.20
HanmanNet-1R	100	100	98.75	100	100	92.20
HanmanNet-1S	98.91	100	98.75	100	100	100
HanmanNet-2A	100	100	98.75	97.72	92.20	100
HanmanNet-2V	98.91	100	100	100	100	92.20
HanmanNet-2G	98.91	97.36	98.75	100	92.20	100

From Table 1, it can be seen that ResNet (M), HanmanNet-1R, and HanmanNet-2V provide 100% accuracy for the two-class classifications involving normal and abnormal mammograms whereas 98.91% classification accuracy is achieved with HanmanNet-1S, HanmanNet-2V, and HanmanNet-2G. Next, the classification of abnormal mammograms into benign and malignant by both HanmanNet-2G and ResNet (M) yields 97.36% accuracy whereas HanmanNets (1R, 1S, 2A, 2G) provide 100% accuracy.

We attempt at two types of three-class classifications of mammograms. The first type is based on the characteristics of the background tissues comprising Fatty, Fatty glandular, and Dense glandular (F/G/D). In this case, 176 mammograms are divided into training to the testing ratio of 3:1. Table 1 shows the results of three-class classifications wherein the features from HanmanNets-(1R, 1S) and HanmanNets-(2A, 2G) provide 98.75% classification accuracy whereas HanmanNet-1R and HanmanNet-2V provide 100% accuracy. The second type of three-class classifications involves normal, benign and malignant (N/B/M) classes. In this case, 320 samples are used in the training-to-testing ratio of 3:1. Here, ResNet (M) and HanmannNet-1R provide 97.72% accuracy whereas HanmanNets-(1R, 1S, 2V, 2G) provide 100% classification accuracy.

In the multi-class classification, the abnormal mammograms are categorized as per the abnormality such as calcification (CALC), well-defined mass (CIRC), spiculated mass (SPIC), ill-defined mass (MISC), architectural distortion (ARCH), and asymmetry (ASYM). For the six-class classifications, we have used 91 abnormal mammograms in the training to the testing ratio of 6:1. The seven-class case has the same mammograms as used in the six-class in addition to the normal mammograms. In this case, 91 mammograms are used for training and 13 for testing.

For the six-class classifications, ResNet (M) and HanmanNets-(1R, 1S, 2V, 2G) give 100% classification accuracy whereas HanmanNet-2A gives 92.20% accuracy but for the seven-class classifications, HanmanNets-(1S, 2G, 2V) give 100% classification accuracy whereas 92.30% classification accuracy is achieved with ResNet (M), and HanmanNets (1R, 2A).

Table 2 provides the classification accuracies achieved by Hanman Transform Classifier on the features extracted from the conventional deep learning architectures. From this table, it can be observed that AlexNet, VGG-16, GoogLeNet and ResNet-18 provide 97.82%, 96.73%, 95.65% and 98.91% accuracies respectively in the classification of mammograms into normal and abnormal. In the next level classification of abnormal mammograms into benign and malignant subclasses, both AlexNet and VGG-16 give 94.73% whereas GoogLeNet and ResNet-18 give 92.30% and 98.91% accuracies respectively. In the three-class classification of mammograms into fatty, fatty glandular and dense glandular; AlexNet and ResNet-18 give 98.75% accuracy whereas VGG-16 and GoogLeNet give 97.5% and 96.25% classification accuracies respectively. ResNet-18 and VGG-16 yield the same accuracy of 97.72% in the three-class classification of mammograms into normal, malignant and benign whereas AlexNet and Goog-

LeNet yield 95.45% and 93.18% accuracies in the same classification. AlexNet gives 84.61% accuracy whereas VGG-16 gives 92.20% accuracy for both the six-class and seven-class classifications. VGG-16 and GoogLeNet provide 84.61% and 76.92% accuracies respectively for the six-class classifications and the corresponding accuracies in the case of seven-class classifications are 76.92% and 92.20% [33]. Thus, in comparison to the conventional deep learning architectures, the proposed architectures perform extremely well.

Table 2. Classification accuracies achieved by the Hanman Transform Classifier (in percentage) for the multi-class classifications using the features of deep learning architectures [33].

Architecture	Normal/ Abnormal	Benign/ Malignant	F/G/D	N/B/M	Six class	Seven Class
AlexNet	97.82	94.73	98.75	95.45	84.61	84.61
VGG-16	96.73	94.73	97.5	97.72	84.61	76.92
GoogLeNet	95.65	92.30	96.25	93.18	76.92	92.20
ResNet-18	98.91	98.75	98.75	97.72	92.20	92.20

Table 3 compares the classification accuracies obtained with the features of HanmanNet-1R with the state-of-the-art classifiers for the two-class classification (Normal/Abnormal). The artificial neural network gives 93.90% accuracy [34] with the textural energy features whereas the support vector machine (SVM) classifier gives 83.87% classification accuracy [35] with the shape and texture features. Hanman Transform Classifier gives 100% classification accuracy for the two-class classification of HanmanNet-1R features. It is higher than that of other classifiers taken for comparison from the literature.

Table 3. A comparison of the classification accuracies for the two-class classifications (Normal, Abnormal).

S. No	Author	Year	Classifier and features	Results
1	Setiawan *et al.* [34]	2015	Artificial neural network applied on the textural energy features	93.90%
2	Soulami *et al.* [35]	2017	SVM classifier applied on shape and texture features	83.87%
4	Proposed Approach	Present	Hanman Transform Classifier applied on the features of HanmanNet-1R	100%

Table 4 shows a comparative analysis of the results of the two-class (Benign, Malignant) classification. Fisher's linear discriminant analysis gives 94.57% accuracy on the linear binary pattern and neighbors structural similarity-based features [36] whereas the parasitic learning networks and artificial neural network-based classifiers give 96.7% accuracy on the CNN features [37]. Note that 95.2% classification accuracy is achieved on the textural features with a neural

network-based classifier [38]. The logistic classifier applied to the statistical texture features gives 96.7% accuracy [39]. The Hanman Transform Classifier performs better by providing 100% accuracy on the features of HanmanNet-1R.

Table 4. A comparison of the classification accuracies for the two-class classifications (Benign, Malignant).

S. No	Author	Year	Classifier and features	Results
1	Rabidas *et al.* [36]	2016	Fisher's LDA (linear discriminant analysis) applied on the neighbor structural similarity and linear binary pattern-based features	94.57%
2	Jiao *et al.* [37]	2018	Parasitic learning network and artificial neural network-based classifier applied on the convolutional neural network (CNN) based features	96.7%
3	Abdelsamea *et al.* [38]	2019	Neural network applied on the texture features	95.2%
4	Boudraa *et al.* [39]	2020	Simple logistic classifier applied on statistics texture-based features	96.7%
5	Proposed approach	Present	Hanman Transform Classifier on features of HanmanNet-1R	100%

Table 5 compares the classification accuracies of the proposed approach with MA-CNN (Multiscale all convolutional neural network) applied to multiscale features [40]. The Hanman Transform Classifier gives the same 100% on specificity, sensitivity, accuracy, along with an F-score of 1 on the features due to HanmanNet-1R. These are higher than those of MA-CNN that provides specificity = 96%, sensitivity = 96%, accuracy = 96.47% along with F-score = 0.97.

Table 5. A comparison of the classification results for the three-class classification (Benign, Malignant, Normal).

S. No	Author	Year	Classifier and Features	Results
1	Agnes *et al.* [40]	2020	MA-CNN (Multiscale all convolutional neural network) applied on multiscale features	specificity = 96%, accuracy = 96.47%, F-score = 0.97, sensitivity = 96%.
2	Proposed Approach	2024	Hanman Transform Classifier applied on the features of HanmanNet-1R	specificity = 100%, accuracy = 100%, F-score = 1, sensitivity = 100%.

Table 6 compares the classification accuracies achieved by the different approaches in the literature for the three-class classifications (fatty, fatty glandular, and dense glandular).

Table 6. A comparison of the classification accuracies for the three-class classification (Fatty, fatty glandular, dense).

S. No	Author	Year	Classifier and features	Results
1	Mustra *et al.* [41]	2012	K-nearest neighbor applied on gray level co-occurrence matrix (GLCM) based features	Accuracy = 82.5%
2	Abdel Nasser *et al.* [42]	2015	SVM classifier applied on ULDP (Uniform local directional pattern) features	Accuracy = 85.5%
3	Arefan *et al.* [43]	2015	ANN classifier applied on the statistical features	Accuracy = 97.66%
4	Nithya *et al.* [44]	2017	ANN classifier applied on the linear binary pattern, histogram, trace transform, Gabor wavelet, GLCM, GLDM, and GLRLM features	sensitivity = 97.5133% specificity = 98.75% Accuracy = 97.5%
5	Proposed approach	2024	Hanman Transform Classifier applied on the features of HanmanNets-1R	accuracy = 98.75% sensitivity = 98.72% specificity = 99.22%

The K-nearest neighbor gives 82.5% classification accuracy on the gray level co-occurrence matrix [41]. The SVM classifier yields 85.5% accuracy on the Uniform Local Directional Pattern (ULDP) features in [42] whereas ANN classifier achieves 97.66% classification accuracy on the statistical features [43]. ANN classifier applied on the histogram, trace transform, Gray Level Run Length Matrix (GLRLM), Gray Level Co-occurrence Matrix (GLCM), linear binary pattern Gabor wavelet, and Gray Level Difference Matrix (GLDM) features provides 97.5% accuracy, specificity of 98.75%, and sensitivity of 97.5133% [44]. The Hanman Transform Classifier provides the classification accuracy of 98.75%, specificity of 99.22%, and sensitivity of 98.75% on the features of HanmanNet-1R.

Table 7 shows a comparison of the classification accuracies for the six-class and seven-class classifications. MM-ANFIS (Memetic meta-heuristic adaptive neuro-based fuzzy inference system) classifier gives 82% accuracy for the six-class classifications and 82.56% accuracy for the seven-class classifications on GLCM descriptors, Zernike moments and 2D wavelet transform features. Compared to this, the Hanman Transform Classifier provides 100% classification accuracy for both the six-class and seven-class classifications on the features from HanmanNet-1S.

Table 7. A comparison of the classification accuracies for the multi-class classifications (CALC/CIRC/SPIC/MISC/ARCH/ASYM).

S. No	Author	Year	Classifier and Features	Results
1	Rezaee, *et al.* [45]	2020	MM-ANFIS classifier applied to GLCM descriptors, 2D wavelet transform, Zernike moments features. Simulated annealing is used for feature selection.	82% (Six-class) 82.56% (seven-class)
2	Proposed Approach	2024	Hanman Transform Classifier applied on the features extracted with Sigmoid ResNet (M) architecture	100% (Six-class) 100% (Seven-class)

7. Conclusions

In view of the widespread use of deep learning neural networks for image processing problems, an attempt is made to study these networks and to address their limitations. As a result of this study, it has been observed that there is a close resemblance between how the features are found from deep learning architectures and the way the information set features are extracted from the input images. Enthused by this analogy, we have proposed two types of HanmanNets that allow us to modify the existing deep learning architectures and help us produce more effective features.

In the CNN architectures, the filters modify the input images at the convolutional layers to yield feature maps. Their sizes are reduced at the pooling layers. The filter action on a subimage turns out to be computing its membership function value. At the selected layer in the case of ResNet (M) and the last layer in the case of CNN architectures, we tap a set of feature maps with their elements resembling the membership function values according to the information set concept. Using this observation, ResNet (M) paves the way for the features of Type-1 HanmanNet-1R, 1S whereas the CNN architectures, viz., AlexNet, GoogLeNet, and VGG-16 pave the way for the features of Type-2 HanmanNets-2A, 2G, 2V.

We have drastically reduced the layers of ResNet-18 by cutting down some of its layers using a skip connection facility while modifying it into ResNet (M). This facility has not been availed in the CNN architectures as it requires a lot of experimentation. This is a boon of the HanmanNets that can be used to simplify any deep learning neural network architecture as a future study. We have derived the mathematical expressions for the conditional divergent information set and Hanman Transform Classifier all of which help in the multi-class classification of mammograms.

It is found that features from HanmanNets are better than the existing CNN features in the literature. The results obtained on the mini MIAS database are validated clinically by the expert radiologists.

The limitations of the present study include: 1) Choice of an appropriate deep learning network architecture for the conversion of its feature maps into the deep information set features, 2) Selection of the layer to tap the feature map, and 3) Difficulty of which layers to skip. We have used a small dataset for implementing our approach. In the future, we would like to repeat our experimental analysis on a large dataset and hopefully on a private dataset from hospitals. An extensive study of other types of deep learning neural networks has to be conducted to bring them into the fold of information set theory.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] American Cancer Society (2019) Breast Cancer Facts & Figures 2019-2020. https://www.cancer.org/research/cancer-facts-statistics/breast-cancer-facts-figures.html

[2] Simonyan, K. and Zisserman, A. (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. http://arxiv.org/abs/1409.1556

[3] Pan, S.J. and Yang, Q. (2010) *IEEE Transactions on Knowledge and Data Engineering*, **22**, 1345-1359. https://doi.org/10.1109/tkde.2009.191

[4] Yosinski, J., Clune, J., Bengio, Y. and Lipson, H. (2014) How Transferable Are Features in Deep Neural Networks? *NIPS*'14: *Proceedings of the* 27*th International Conference on Neural Information Processing Systems*, Volume 2, 3320-3328.

[5] Chen, Y., Zhang, Q., Wu, Y., Liu, B., Wang, M. and Lin, Y. (2019) Fine-Tuning ResNet for Breast Cancer Classification from Mammography. In: Wu, C., Chyu, M.C., Lloret, J. and Li, X., Eds., *Proceedings of the* 2*nd International Conference on Healthcare Science and Engineering*, Springer, 83-96.

[6] Ting, F.F., Tan, Y.J. and Sim, K.S. (2019) *Expert Systems with Applications*, **120**, 103-115. https://doi.org/10.1016/j.eswa.2018.11.008

[7] Hassan, S.A., Sayed, M.S., Abdalla, M.I. and Rashwan, M.A. (2020) *Multimedia Tools and Applications*, **79**, 30735-30768. https://doi.org/10.1007/s11042-020-09518-w

[8] Rybiałek, A. and Jeleń, Ł. (2020) Application of DenseNets for Classification of Breast Cancer Mammograms. In: Saeed, K. and Dvorský, J., Eds., *Computer Information Systems and Industrial Management*, Springer, 266-277.

[9] Bruno, A., Ardizzone, E., Vitabile, S. and Midiri, M. (2020) *Journal of Medical Signals & Sensors*, **10**, 158-173. https://doi.org/10.4103/jmss.jmss_31_19

[10] Deng, J., Ma, Y., Li, D., Zhao, J., Liu, Y. and Zhang, H. (2020) *Computer Methods and Programs in Biomedicine*, **193**, Article ID: 105489. https://doi.org/10.1016/j.cmpb.2020.105489

[11] Matsuyama, E., Takehara, M. and Tsai, D. (2020) *Open Journal of Medical Imaging*, **10**, 17-29. https://doi.org/10.4236/ojmi.2020.101002

[12] Mohanty, F., Rup, S., Dash, B., Majhi, B. and Swamy, M.N.S. (2020) *Applied Soft Computing*, **91**, Article ID: 106266. https://doi.org/10.1016/j.asoc.2020.106266

[13] Nirmala, G. and Suresh Kumar, P. (2020) *Journal of Ambient Intelligence and Humanized Computing*, **12**, 4797-4808. https://doi.org/10.1007/s12652-020-01890-7

[14] Mohanty, F., Rup, S. and Dash, B. (2020) *Biomedical Signal Processing and Control*, **62**, Article ID: 102108. https://doi.org/10.1016/j.bspc.2020.102108

[15] Muduli, D., Dash, R. and Majhi, B. (2020) *Biomedical Signal Processing and Control*, **59**, Article ID: 101912. https://doi.org/10.1016/j.bspc.2020.101912

[16] Yu, X. and Wang, S. (2019) *Fundamenta Informaticae*, **168**, 219-230. https://doi.org/10.3233/fi-2019-1829

[17] Fu, Y., Patel, B.K., Wu, T., Li, J. and Gao, F. (2020) Advanced Medical Imaging Analytics in Breast Cancer Diagnosis. In: Smith, A., Ed., *Women in Industrial and Systems Engineering*, Springer, 301-319.

[18] Li, H., Zhuang, S., Li, D., Zhao, J. and Ma, Y. (2019) *Biomedical Signal Processing and Control*, **51**, 347-354. https://doi.org/10.1016/j.bspc.2019.02.017

[19] López-Cabrera, J.D., López Rodríguez, L.A. and Pérez-Díaz, M. (2020) *Inteligencia Artificial*, **23**, 56-66. https://doi.org/10.4114/intartif.vol23iss65pp56-66

[20] Suckling, J., Boggis, C.R.M., Hutt, I., Astley, S., Betal, D., Cerneaz, N., Karrsemeijer, N. and Clark, A. (1994) The mini-MIAS Database of Mammograms. In: Suckling, J., *et al.*, Eds., *The Mammographic Image Analysis Society Digital Mammogram Database*, Exerpta Medica, International Congress Series 1069, 375-378. http://peipa.essex.ac.uk/info/mias.html

[21] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) ImageNet Classification with Deep Convolutional Neural Networks. In: *Advances in Neural Information Processing Systems*, Vol. 25, MIT Press, 1097-1105 https://kr.nvidia.com/content/tesla/pdf/machine-learning/imagenet-classification-with-deep-convolutional-nn.pdf

[22] Simonyan, K. and Zisserman, A. (2015) Very Deep Convolutional Networks for Large-Scale Image Recognition. 3*rd International Conference on Learning Representations*, *ICLR* 2015, San Diego, 7-9 May 2015, 1-14. https://arxiv.org/pdf/1409.1556.pdf

[23] Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D., *et al.* (2015) Going Deeper with Convolutions. 2015 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Boston, 7-12 June 2015, 1-9. https://doi.org/10.1109/cvpr.2015.7298594

[24] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/cvpr.2016.90

[25] Razavian, A.S., Azizpour, H., Sullivan, J. and Carlsson, S. (2014) CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. 2014 *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Columbus, 23-28 June 2014, 512-519. https://doi.org/10.1109/cvprw.2014.131

[26] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E. and Darrell, T. (2014) DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. *The* 31*st International Conference on Machine Learning* (*ICML* 2014), Beijing, 21-26 June 2014, 647-655. https://arxiv.org/abs/1310.1531

[27] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Identity Mappings in Deep Residual Networks. In: Leibe, B., Matas, J., Sebe, N. and Welling, M., Eds., *Computer Vision ECCV* 2016, Springer, 630-645.

[28] Khan, A., Sohail, A., Zahoora, U. and Qureshi, A.S. (2020) *Artificial Intelligence Review*, **53**, 5455-5516. https://doi.org/10.1007/s10462-020-09825-6

[29] Hanmandlu, M. and Das, A. (2011) *Defence Science Journal*, **61**, 415-430. https://doi.org/10.14429/dsj.61.1177

[30] Sayeed, F. and Hanmandlu, M. (2017) *Knowledge and Information Systems*, **52**, 485-507. https://doi.org/10.1007/s10115-016-1017-x

[31] Hanmandlu, M., Bansal, M. and Vasikarla, S. (2020) *Journal of Modern Physics*, **11**, 122-144. https://doi.org/10.4236/jmp.2020.111008

[32] Dabass, J., Hanmandlu, M. and Vig, R. (2021) *SN Applied Sciences*, **3**, Article No. 610. https://doi.org/10.1007/s42452-021-04616-2

[33] Dabass, J., Hanmandlu, M. and Vig, R. (2020) *Informatics in Medicine Unlocked*, **20**, Article ID: 100401. https://doi.org/10.1016/j.imu.2020.100401

[34] Setiawan, A.S., Elysia, Wesley, J. and Purnama, Y. (2015) *Procedia Computer Science*, **59**, 92-97. https://doi.org/10.1016/j.procs.2015.07.341

[35] Soulami, K.B., Saidi, M.N. and Tamtaoui, A. (2017) A CAD System for the Detection of Abnormalities in the Mammograms Using the Metaheuristic Algorithm Par-

ticle Swarm Optimization (PSO). In: El-Azouzi, R., Menasche, D.S., Sabir, E., De Pellegrini, F. and Benjillali, M., Eds., *Advances in Ubiquitous Networking* 2, *UNet* 2016, Springer, 505-517.

[36] Rabidas, R., Midya, A. and Chakraborty, J. (2018) *IEEE Journal of Biomedical and Health Informatics*, **22**, 826-834. https://doi.org/10.1109/jbhi.2017.2715021

[37] Jiao, Z., Gao, X., Wang, Y. and Li, J. (2018) *Pattern Recognition*, **75**, 292-301. https://doi.org/10.1016/j.patcog.2017.07.008

[38] Abdelsamea, M.M., Mohamed, M.H. and Bamatraf, M. (2019) *Cancer Informatics*, **18**. https://doi.org/10.1177/1176935119857570

[39] Boudraa, S., Melouah, A. and Merouani, H.F. (2020) *Evolving Systems*, **11**, 697-706. https://doi.org/10.1007/s12530-019-09322-4

[40] Agnes, S.A., Anitha, J., Pandian, S.I.A. and Peter, J.D. (2019) *Journal of Medical Systems*, **44**, Article No. 30. https://doi.org/10.1007/s10916-019-1494-z

[41] Muštra, M., Grgić, M. and Delač, K. (2012) *Automatika*, **53**, 362-372. https://doi.org/10.7305/automatika.53-4.281

[42] Abdel-Nasser, M., Rashwan, H.A., Puig, D. and Moreno, A. (2015) *Expert Systems with Applications*, **42**, 9499-9511. https://doi.org/10.1016/j.eswa.2015.07.072

[43] Arefan, D., Talebpour, A., Ahmadinejhad, N. and Asl, A.K. (2015) *Journal of Instrumentation*, **10**, T12002. https://doi.org/10.1088/1748-0221/10/12/t12002

[44] Nithya, R. and Santhi, B. (2017) *Journal of Instrumentation*, **12**, P07009. https://doi.org/10.1088/1748-0221/12/07/p07009

[45] Rezaee, K., Rezaee, A., Shaikhi, N. and Haddadnia, J. (2020) *SN Applied Sciences*, **2**, Article No. 1297. https://doi.org/10.1007/s42452-020-3103-7