JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

CLASSIFICATION OF RESPIRATORY TRACT DISEASES USING MACHINE LEARNING TECHNIQUES

G. Natarajan

Research Scholar, Department of Computer and Information Science, Annamalai University

natarajang@gmail.com

Dr. P. Dhanalakshmi

Professor, Department of Computer Science and Engineering, Annamalai University

abidhana01@gmail.com

ABSTRACT

Respiratory diseases can affect all structures and organs related to breathing, including the nasal cavity, pharynx (or pharynx), larynx, trachea (or trachea), bronchi and bronchioles, as well as lung tissue and respiratory muscles in the chest. In the proposed work, chest x-ray images are taken in which Histogram of Oriented Gradients and Local Binary Pattern are used for feature extraction, the extracted features are fed into Radial Basis Function Neural Network and Gaussian Mixture Model which was classified into normal, Pleural Effusion, Pneumonia and Tuberculosis. When comparing Histogram of Oriented Gradients with Radial Basis Function Neural Network gives the highest accuracy of 89. 83 %.

Keywords

Radial Basis Function Neural Network (RBFNN), Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP).

INTRODUCTION

Respiratory disease is a type of disease that affects the lungs and other parts of the respiratory system. This can be caused by infections, smoking or inhaling secondhand smoke, radon, asbestos, or other forms of air pollution.

Respiratory diseases include pneumonia, Tuberculosis, Common Obstructive Pulmonary Disease, Pleural Effusion, Asthma, Pulmonary Fibrosis and lung Cancer. Among this, the most common respiratory diseases are Pleural Effusion, Tuberculosis and Pneumonia.

Respiratory system diseases, all respiratory and lung diseases and disorders that affect human breathing (Mitton *et al.*, 2021).

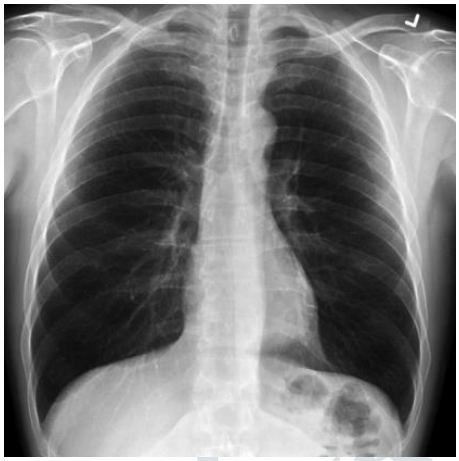
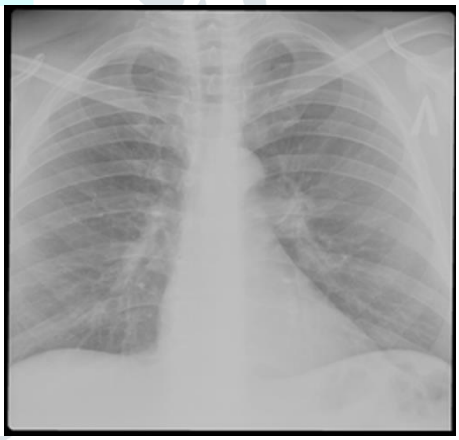


Fig 1 (a) Normal image **b) Pleural Effusion**



c) Pneunomia **d) Tuberculosis**

They include respiratory diseases, including trachea, bronchi, bronchioles, alveoli, pleura, pleural cavity, nerves and respiratory muscles. Respiratory diseases range from mild and self-limiting diseases (such as the common cold, flu, and pharyngitis) to life-threatening diseases (such as pneumonia, pulmonary embolism, tuberculosis, acute asthma, lung cancer (Vos *et al.*, 2021) and severe acute respiratory system Syndrome. (Koegelenberg *et al.*, 2021) Respiratory diseases can be classified in many different ways, including the affected organ or tissue, the type and pattern of signs and symptoms associated with it, or the cause of the disease.

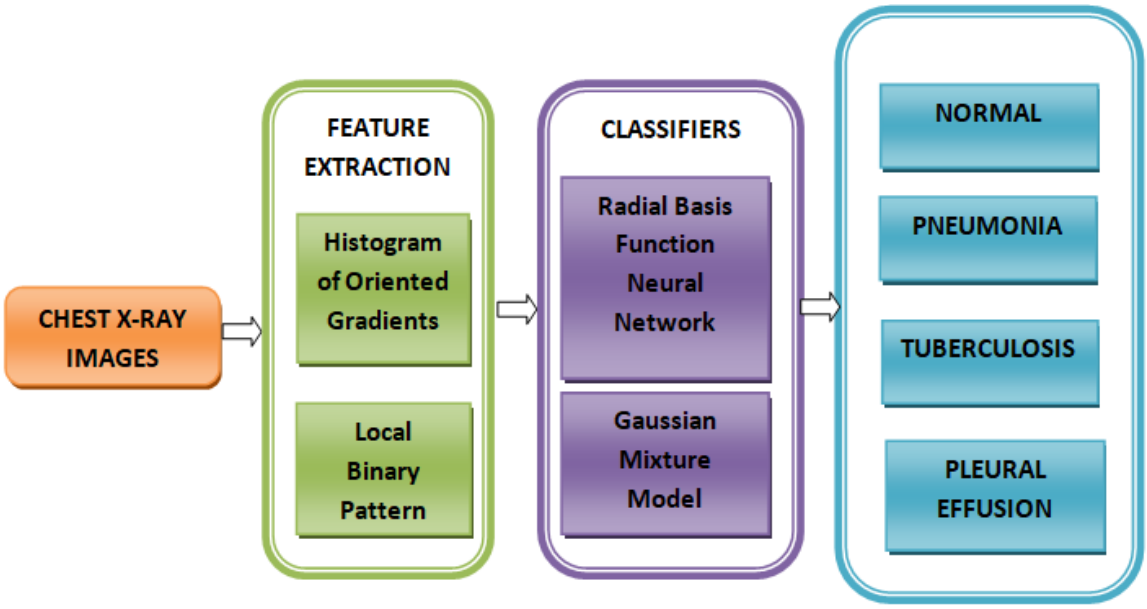


Fig 2. Frame Work of the Proposed System

An acute respiratory infection is an infection that can interfere with normal breathing. It can only affect your upper respiratory system, which starts in your sinuses and ends in your vocal cords, or just your lower respiratory system, which begins in your vocal cords and ends in your lungs. This infection is especially dangerous for children, the elderly, and people with immune system disorders

2. FEATURE EXTRACTION STAGE

2.1 Histogram of Oriented Gradients

The HOG method is specially designed for the identification of objects. The main application is to generate the image-oriented gradient histogram, which describes the distribution of intensity gradients or edge directions. Here the image is divided into small areas called cells and a gradient direction histogram is created for each pixel within the cell.(Kaushik *et al.*, 2019).

Then it concatenates the histogram and normalizes it by calculating the intensity values over the largest area of the image, this is called a block. In this way we can normalize all cells within the block (Stahl *et al.*, 2008). Normalization represents a change in illumination and shading. The implementation of the HOG algorithm involves the following steps

- Gradient Computation
- Histogram Generation
- Block Normalization

Gradient Computation

Fig.3shows the concept of cells and blocks used for the extraction of HOG properties. The cell size is generally 8x8 pixels. A block is created by grouping the 2x2 cells. In the first step, the HOG characteristics are extracted to determine the difference values, and for the x and y directions it is calculated by the following equation

$$\begin{cases} fx(x,y) = f(x+1,y) - f(x-1,y) \\ fy(x,y) = f(x,y+1) - f(x,y-1) \end{cases} \quad (1)$$

Where $f(x,y)$ is the brightness value of the image in (x,y)

The calculation of magnitude *Arg* and direction θ in (x,y) takes place using the Eqs. (2) and (3) respectively.

$$arg(x,y) = \sqrt{f_x(x,y)^2 + f_y(x,y)^2} \quad (2)$$

$$\theta(x,y) = arctan\frac{f_{x(x,y)}}{f_{y(x,y)}} \quad (3)$$

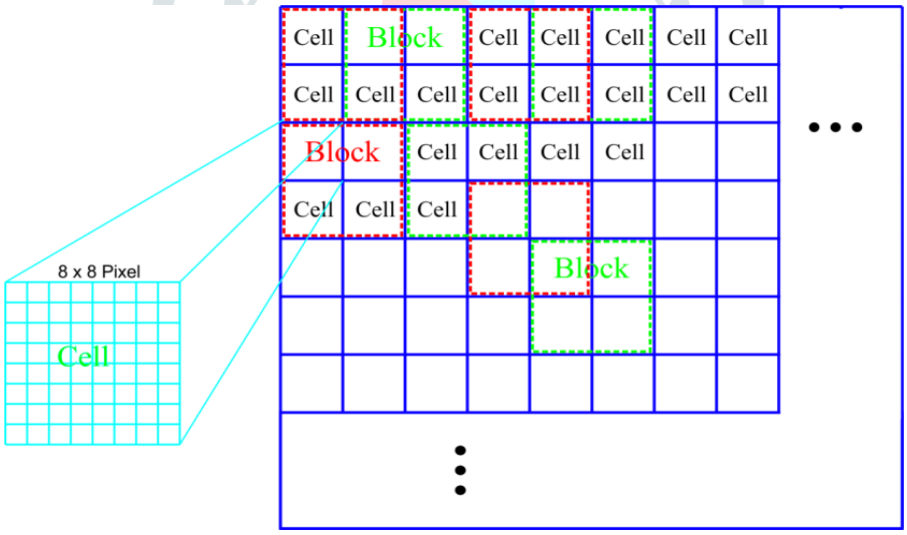


Fig. 3 Cells and Blocks in HOG

Histogram Generation

Based on the size of the gradient and the previously calculated direction of the gradient, the histogram is created for each cell. As shown in Figure 3,the gradient orientation is segmented as a container in a cell and a total of 9 containers of 4 cells are separated by a block and accumulated independently to generate four 9-dimensional feature vectors which are integrated into a 36-dimensional feature vector (Wu *et al.*, 2019).

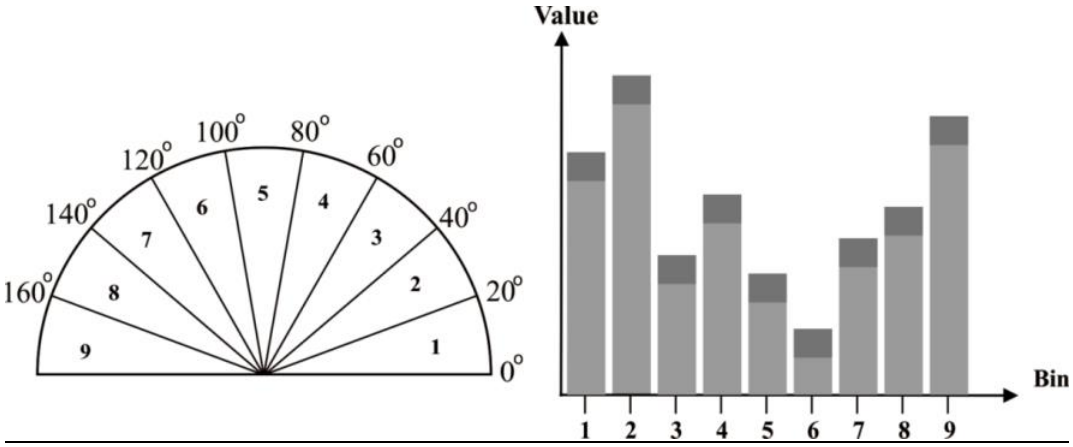


Fig. 4 Orientation Bins in HOG

For reducing the aliasing effect, weighted gradient values are interpolated bi-linearly among the nearby bin centers in position.

Block Normalization

Due to the variations in the transformation of the illumination conditions and the background of the image, the range of the gradient value is wider, so better normalization is very important to see the features. The normalization process divides the feature vector v by L2norm,

$$\upsilon \rightarrow \frac{\upsilon}{\sqrt{||\upsilon||^2_{2+\varepsilon^2}}} \quad (4)$$

where ε is a small constant representing divisions by zero and v is a normalized vector.

3.2 Local Binary Pattern

Another feature extraction technique commonly used to filter textures and patterns in a delivered image is the LBP model. It determines the local representation of the texture that occurs by comparing each pixel with the values of neighboring pixels (Fradi *et al.*, 2013). The steps of the LBP method are

Algorithm of LBP

- The first step lies in constructing LBP texture descriptor to convert the image to gray scale.
- LBP descriptor operates on a fixed 3 x 3 neighborhood of pixels.
- We consider the center pixel and threshold it against its neighborhood of 8 pixels.

- If the intensity of the center pixel is more-than-or-equal to its neighbor, then we maintain the value to 1, or we keep it to 1.
- The LBP value for the center pixel either in clockwise or in counter-clockwise direction is computed.
- For 3 x 3 neighborhood, there are 8 neighbors and the binary test is performed.
- The results of the binary test are saved in 8-bit array, and then we convert to decimal value.
- LBP is considered to be uniform if it contains at most 0-1 or 1-0 transitions.
- The process of thresholding, gathering binary strings and saving the output decimal value in the LBP array is repeated for each pixel in the input image.
- Next, histogram over the output LBP array is computed.
- 3 x 3 neighborhoods contain $2^8 = 256$ possible patterns, thus a minimum value is 0 and a maximum value is 255.
- So, we construct a 256-bin histogram of LBP.

3 MODELING THE FEATURES

3.1 Radial Basis Function Neural Network

Radial Basis Function Neural Network is a feed forward structure with an input layer, a hidden layer and an output layer. There is a two layer feed forward network during which the radial basis functions are embedded. A set of inputs and output characterize the network. There are hidden units between the inputs and outputs. These hidden units responsible for implementing the radial basis function (Maeda *et al.,* 2006).

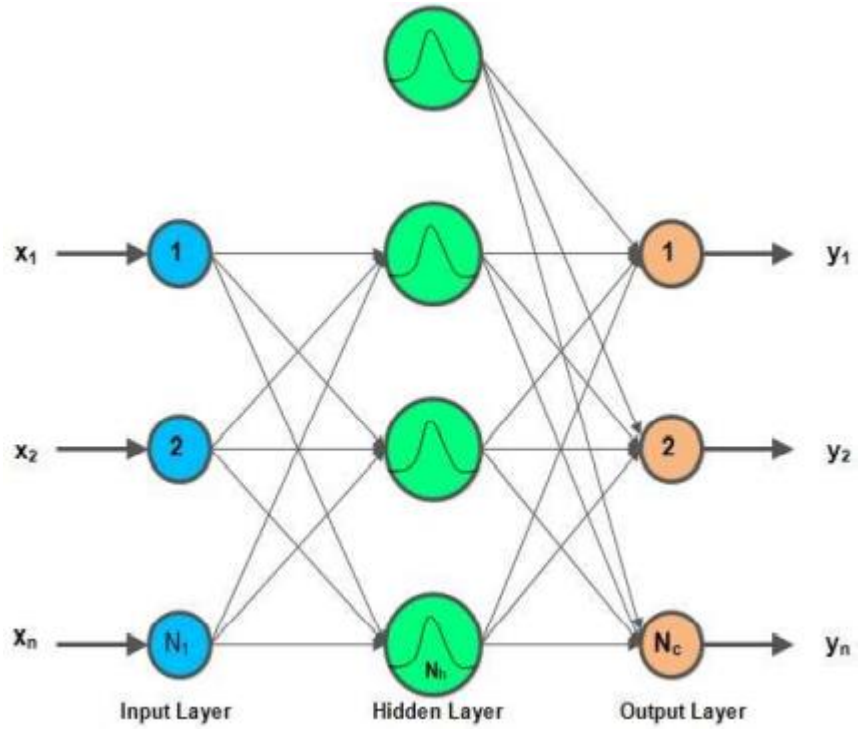


Fig 5 Radial Basis Function Neural Network

There are Ni units in the input layer for the Ni dimensional input vector. N_h hidden layers are connected to these input units and are finally connected to these Nc output layer units, and Nc denotes the total of output classes. The activation functions of these hidden layers are Gaussians (Erol *et al.*, 2008) mean vector(centers) characterize such functions along with the covariance matrices. The activation function of the i^{th} hidden unit for an input vector x_i is given by.

$$g_i(\boldsymbol{x_j}) = exp\left(\frac{-\|\boldsymbol{x_j} - \boldsymbol{\mu_i}\|^2}{2\sigma_i^2}\right) \quad (5)$$

suitable clustering algorithm calculates the μ_i and σ_i^2. The K means clustering algorithm is used for the determination of the centers. Following steps compose the algorithm.

1. Initialize the samples randomly to k mean values (clusters) $\boldsymbol{\mu_1, \cdots, \mu_k}$.
2. Classify n samples according to nearest $\boldsymbol{\mu_k}$
3. Recompute $\boldsymbol{\mu_k}$
4. Repeat the step 2 and 3 until no change in $\boldsymbol{\mu_k}$

The number of activation functions within the network and their spread influence the smoothness of the mapping. The assumption $\sigma_i^2 = \sigma^2$ is made and σ^2 is given in (6) to ensure that the activation functions are not too peaked or too flat

$$\sigma^2 = \frac{\eta d^2}{2} \tag{6}$$

where d is that the maximum distance between the chosen centers, and η is an empirical scale factor which serves to control the smoothness of the mapping function. Therefore, equ 6 can be written as

$$g_i(\boldsymbol{x_j}) = exp\left(\frac{-\|\boldsymbol{x_j} - \boldsymbol{\mu_i}\|^2}{\eta d^2}\right) \tag{7}$$

The hidden layer units are fully connected to the N_c output layer units through weights w_{ik}. The output units are linear, and the response of the k^{th} output unit for an input x_j is given by

$$y_k(\boldsymbol{x_j}) = \sum_{n_h}^{i=0} w_{ik} g_i(\boldsymbol{x_j}), k = 1, 2, \cdots, n_c \tag{8}$$

where $g_0(x_j) = 1$. Given nt feature vectors from n_c classes, training the RBFNN involves estimating μ_i , i = 1, 2, · · · ,n_h, η, d2 , and w_{ik}, i = 0, 1, 2, · · · ,n_h, k = 1, 2, · · · ,n_c. The training procedure is given below

Determining μi and d2: Traditionally, the unsupervised k-means clustering algorithm (Duda*et al.*, 2001) can be used to find N_h clusters from the N_t training vector. However, we cannot classify a class training vector into a single cluster. To get clusters by class only, we can use k-means clustering in a supervised way.

The training feature vector belonging to the same class was grouped into clusters N_h/N_c using K-means clustering algorithm. This is repeated for each layer creating a cluster of N_h for the N_c layers. These cluster means are used as the center μ_i of the Gaussian activation functions in RBFNN. The parameter d is then calculated to be by finding the largest distance between the clusters that means N_h.

Determining the weights w_{ik} between the hidden and output layer: Since centers and widths of Gaussian functions are calculated from the training vectors N_t, (8) can be written in matrix as follows

$$\mathbf{Y} = \mathbf{GW} \tag{9}$$

where Y is a $N_t \times N_c$ matrix with elements $Y_{ij} = y_j(x_i)$, G is a $N_t \times (N_h + 1)$ matrix with elements $G_{ij} = y_j(x_i)$, and W is a $(N_h + 1) \times N_c$ matrix of unknown weights. W is obtained from the standard least squares solution as given by

$$\mathbf{W} = (\mathbf{G}^{\mathbf{T}}\mathbf{G})^{-1}\mathbf{G}^{\mathbf{T}}\mathbf{Y} \tag{10}$$

To solve W from (10), G is completely specified by the clustering results, and the elements of Y are specified as

$$Y_{ij} = \begin{cases} 1, & \text{if } \boldsymbol{x_i} \in \ class\ j, \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

3.2 Gaussian Mixture Models

The probability distribution of the feature vectors is modeled by parametric or non-parametric methods. Models in the form of the probability density function are called parametric. In case of non-parametric modeling, minimal or no assumptions are made with regard to the probability distribution of the feature vectors (Shahin *et al.*, 2019).

The potential of Gaussian mixture models to represent an underlying set of Chest X-ray classes by individual Gaussian components in which the spectral shape of the chest x-ray class is parameterized by the mean value vector and the covariance matrix is significant. In addition, these models have the ability to arbitrarily approximate the observation densities in the absence of other information (Ma *et al.*, 2019).With mixed Gaussian models, each radiograph is modeled as a mixture of several Gaussian groups in feature space. The basis for using GMM is that the distribution of feature vectors extracted from a class can be modeled from a mixture of Gaussian densities as shown in Fig.6

For a D dimensional feature vector x, the mixture density function for category s is defined as

$$p(\boldsymbol{x}/\lambda^s) = \sum_{i=1}^{M} \alpha_i^s f_i^s(\ \boldsymbol{x}) \tag{12}$$

The mixture density function is a weighted linear combination of m component uni-modal Gaussian densities $f_i^s(.)$.

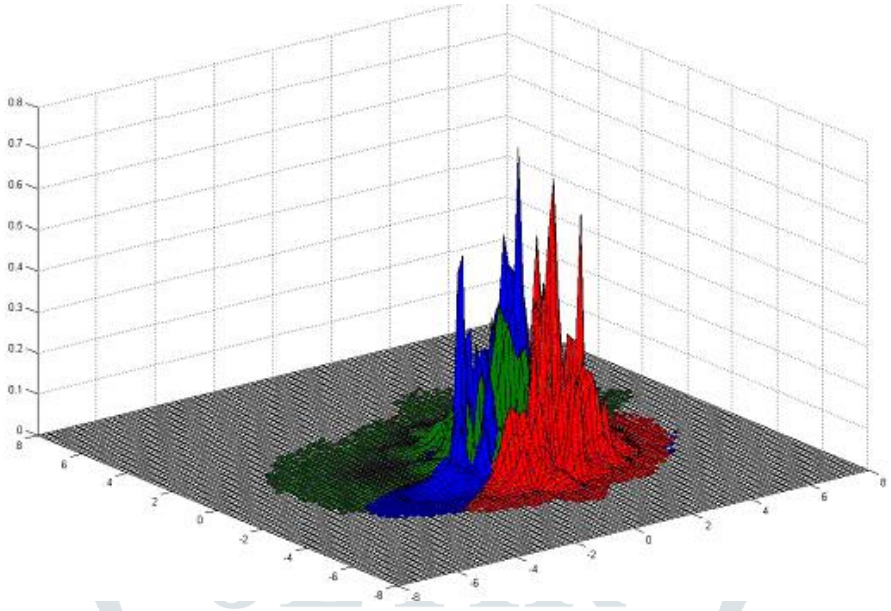


Fig. 6 Gaussian mixture models

Each Gaussian density function $f_i^s(.)$ is parameterized by the mean vector $\mu^s{}_i$ and the covariance matrix $\Sigma^s{}_i$ using

$$f_i^s(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^d|\boldsymbol{\Sigma}_i^s|}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i^s)^T(\boldsymbol{\Sigma}_i^s)^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i^s)\right), \quad (13)$$

where $(\Sigma^s{}_i)^{-1}$ and $|\Sigma^s{}_i|$ denote the inverse and determinant of the covariance matrix $\Sigma^s{}_i$, respectively. The mixture weights ($\alpha^s{}_1, \alpha^s{}_2, ..., \alpha^s{}_M$) satisfy the constraint $\sum_{i=1}^{M} \alpha_i^s = 1$. Collectively, the parameters of the model λ^s are denoted as $\lambda^s = \{\alpha^s{}_1, \alpha^s{}_2, ..., \alpha^s{}_M\}$, i = 1, 2, · · ·,M. The number of mixture components is chosen empirically for a given data set. The parameters of GMM are estimated using the iterative expectation maximization algorithm

The motivation for using Gaussian densities to represent chest x-ray image features is the potential of GMMs to represent an underlying set of respiratory tract disease classes by individual Gaussian components, in which the spectral shape of the respiratory tract disease class is parameterized by the mean vector and the covariance matrix. In addition, GMMs have the ability to estimate observation densities in the absence of other information. (Das *et al.,* 2019). In GMMs, each chest x-ray image is modeled as a mixture of several Gaussian groups in the feature space.

4 PERFORMANCE MEASURES

To evaluate the performance of GMM classifier using HOG and LBP feature extraction techniques, a set of evaluation parameters namely Accuracy, Precision, Recall and F-score are used in this work.

Accuracy

The accuracy of a measurement system is a level of measurement that yields are true (no systemic errors) and consistent (no random errors) results.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (14)$$

Precision

In classification work, the precision for the class is the number of TP divided by the total number of elements labeled as belonging to the positive class

$$Precision = \frac{TP}{TP + FP} \quad (15)$$

Recall

Recall is defined as the number of TP divided by the total number of elements that actually belong to the positive class

$$Recall = \frac{TP}{TP + FN} \quad (16)$$

F-Measure

F-Measure is a measure of test's accuracy and it takes into account of both the precision and recall of the test to compute the score.

$$F - Measure = 2\frac{Precision * Recall}{Precision + Recall} \quad (3.17)$$

5 EXPERIMENTAL RESULTS

5.1 Dataset

The dataset are collected from NIH database. A total of 600 X-ray samples were collected from online database from different patients. 400 images were used for training and 200 were used for testing. In this 400 images (100

– Pneumonia, 100 – Pleural Effusion, 100 – Tuberculosis, 100 – Normal). In this 200 testing images (50 – Pneumonia, 50 – Pleural Effusion, 50 – Tuberculosis, 50 – Normal).

5.2.1 Feature Extraction Using HOG

HOG features were extracted from respiratory tract images affected images. The input image (128 x 128) is now given as the input to HOG. In the first step, gradient values were calculated along the horizontal and vertical directions. The Sobel filter mask is applied on the image. The descriptor blocks normalize all the histogram in the image. Each area is split into 4 x 4sub-regions where each region produces 9 bin gradient orientation. An 8100-dimensional feature vector is obtained using HOG for a single image. 400 images were used for training and hence we arrived at a 400 x 8100 matrix.

5.2.2 Feature Extraction using LBP

LBP features were extracted from respiratory tract images. The input image (128 x 128) is now given as the input to LBP. In the first step, LBP texture descriptor transforms the image into gray scale and operates on a stable 3 x 3 neighborhood of pixels. Next, the center pixel is taken and threshold against its neighborhood of 8 pixels. If the intensity of the center pixel is more-than-or-equal to its neighbor, we define the value to 1, otherwise we set to 0. The binary test is performed for 8 neighbors and the value is stored in 8 -bit array which is converted into decimal value. LBP performs both the uniform and non-uniform transitions. A 256-bin histogram is computed over the LBP array. A 59- dimensional feature vector is extracted using LBP for a single image. 400 images were used for training and hence we arrived at a 400 x 59 matrix.

5.2.3 Evaluation using RBFNN

For RBFNN training 8100 dimensional HOG features and 59-dimensional features are extracted from the chest x-ray images for each category. The features are given as the input to the RBFNN model. The RBF centres are located using K-means algorithm. The weights are determined using K-means algorithm. Experiments are conducted by varying the values of k = 1, 2, 3 and 5 for each category. Table 1 and Fig. 7 shows the performance of RBFNN with HOG and LBP features.

Table 1 Performance of RBFNN

Features	Performance (in %)
HOG	89.83
LBP	89.75

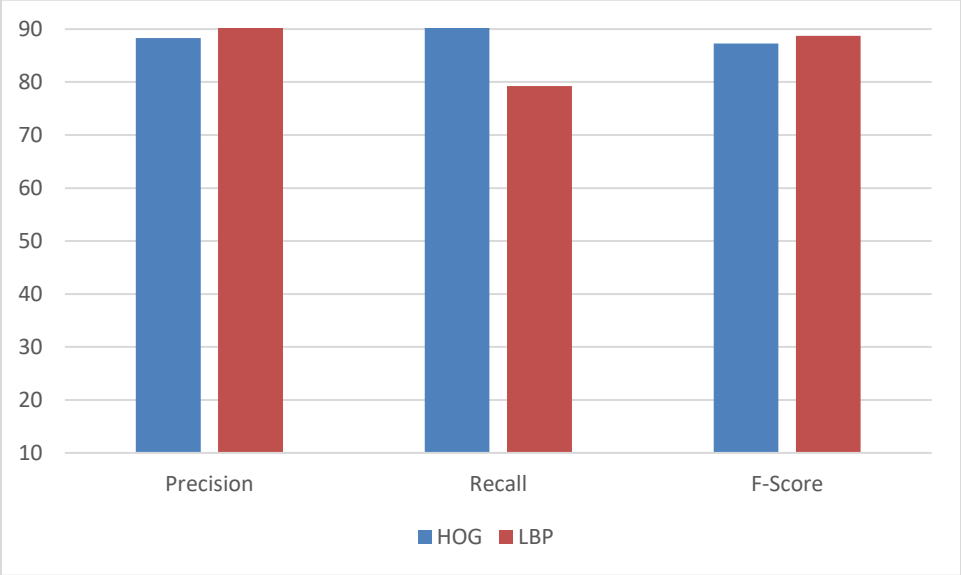


Fig. 7 Performance of RBFNN with HOG and LBP

5.2.4 Evaluation using GMM

In GMM the database comprises of chest x-ray samples that leads to fitting of each category to individual component. The component setting of 4 or more provide better accuracy than others. Based on the characteristics of each disease the sub categories are analyzed.

Table 2 Performance of GMM for various mixtures

No. of Mixtures	2	4	6
HOG	81.18	84.71	84.81
LBP	80.11	85.32	85.50

Various components in GMM using HOG and LBP features are analyzed and are shown in Table 3.5. The number of Gaussian mixtures is increased from 2 to 10 and the performance in terms of classification accuracy is studied. When the number of mixtures is 2, the performance is very low. When the mixtures are increased from 2 to 4, the classification performance slightly increases. When the number of mixtures varies from 4 to 10, there is no considerable increase in the performance and the maximum performance is achieved. There is no considerable increase in the performance when the number of mixtures is above 10. With GMM, the best performance is achieved with 4 Gaussian mixtures. Performance of GMM using HOG and LBP features with 4 Gaussian mixtures is discussed in Section 3.2.

5.2.6 Performance of Detecting Normal/ Abnormal Respiratory Tract Diseases

For GMM 4 Gaussian mixture model provides the better performance when LBP features are given to GMM. From the overall analysis, GMM with LBP shows the average accuracy of 85.5%. The performance of Detecting Normal/Abnormal Respiratory Tract Diseases is shown in Table 3.3.

Table 3 Performance of GMM with HOG and LBP features

Classifiers		**GMM**		
Performance(in %)		**Precision**	**Recall**	**F-Score**
HOG	2	81.40	92.31	86.50
	4	82.71	85.11	89.77
	6	91.32	86.50	89.82
LBP	2	80.30	89.41	85.31
	4	83.81	90.20	86.88
	6	90.30	88.80	89.54

5.2.7 Comparative Result Analysis

This section made a comparative study of two different feature extractions namely, HOG and LBP. The classifiers are RBFNN and GMM. The results are provided in Table 3.1 and also in Fig.3.2. From the Fig. 3.7 the results show that RBFNN with HOG achieved comparably better results when compared with other techniques. It is noted that the highest accuracy of 89.83 % is obtained with RBFNN using HOG features.

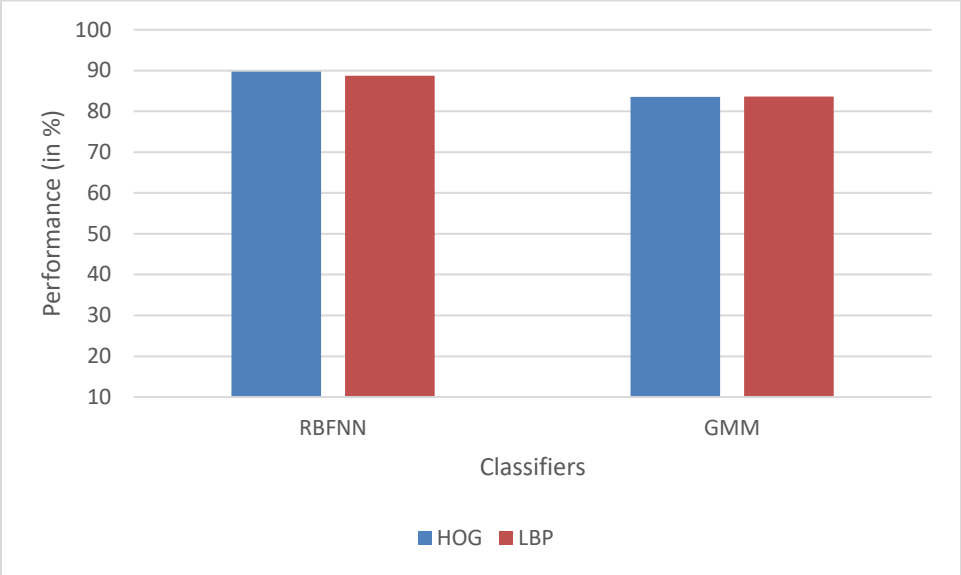


Fig. 8 Overall Performance of RBFNN and GMM

6 Conclusion

From the above experiments various components in RBFNN and GMM using HOG and LBP features is analyzed. From the experiments, it is found when the number of mixtures varies from 2 to 10, there is no considerable increase in the performance and the maximum performance is achieved. There is no considerable increase in the performance when the number of mixtures is above 10. With RBFNN, the best performance is achieved with 4 Gaussian mixtures. The highest accuracy that has been achieved is 89.83% HOG features are fed into RBFNN. With GMM, the best performance is achieved with 4 Gaussian mixtures. By comparing RBFNN with HOG features gives better performance than other results.

References

1. **Vos, L. M., Bruyndonckx, R., Zuithoff, N. P. A., Little, P., Oosterheert, J. J., Broekhuizen, B. D. L and GRACE, (2021),** Consortium. Lower respiratory tract infection in the community: associations between viral a etiology and illness course, *Clinical Microbiology and Infection*, Vol 27(1), pp. 96-104.
2. **Mitton, B., Rule, R., and Said, M., (2021),** Laboratory evaluation of the BioFire Film Array Pneumonia plus panel compared to conventional methods for the identification of bacteria in lower respiratory tract specimens: a prospective cross-sectional study from South Africa, *Diagnostic Microbiology and Infectious Disease*, Vol 99(2).

3. **Koegelenberg, C. F., Schoch, O. D., and Lange, C, (2021),** Tuberculosis: The Past, the Present and the Future. *Respiration*, Vol 100(7), pp. 553-556.
4. **Kaushik, D., Hovy, E., & Lipton, Z. C (2019),** Learning the difference that makes a difference with counterfactually-augmented data, *ICLR 2020 Conference.*
5. **Stahl, H., Schädler, K., and Hartung, E, (2008),** Capturing 2D and 3D biometric data of farm animals under real-life conditions. *In Proceedings in international conference of agricultural engineering*,Vol. 1034.
6. **Fradi, H., and Dugelay, J. L,(2013),**A new multiclass SVM algorithm and its application to crowd density analysis using LBP features, *International Conference on Image Processing, IEEE*, pp. 4554-4558.
7. **Ma, J., Jiang, X., Jiang, J., and Gao, Y (2019),** Feature-guided Gaussian mixture model for image matching. *Pattern Recognition*, Vol 92,pp. 231-245.
8. **Shahin, I., Nassif, A. B., and Hamsa, S, (2019),** Emotion recognition using hybrid Gaussian mixture model and deep neural network, *IEEE access*, Vol 7, pp. 26777-26787.
9. **Wu, Z., Su, L., and Huang, Q (2019),** Cascaded partial decoder for fast and accurate salient object detection. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp.
10. **Das, A., Acharya, U. R., Panda, S. S., and Sabut (2019),**Deep learning based liver cancer detection using watershed transform and Gaussian mixture model techniques, *Cognitive Systems Research*, Vol 54, pp. 165-175.
11. **Maeda, K., Kanae, S., Yang, Z. J., and Wada, K. (2006),** Design of RBF network based on fuzzy clustering method for modeling of respiratory system. In *International Symposium on Neural Networks* (pp. 746-753). Springer, Berlin, Heidelberg.
12. **Erol, R., Oğulata, S. N., Şahin, C., and Alparslan, Z. N. (2008),** A radial basis function neural network (RBFNN) approach for structural classification of thyroid diseases. *Journal of medical systems*, *32*(3), 215-220.
13. **Duda,R. O., Hart, P. E., and Stork, D. G, (2001),** Pattern Classification, *John Wiley Interscience*, New York,.