ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2019.0285

A STRATEGY FOR CONTENT BASED IMAGE RETRIEVAL AND FOREST FIRE DETECTION FROM REMOTELY SENSED IMAGES

C. Jenifer Grace Giftlin and S. Jenicka
Department of Computer Science and Engineering, Sethu Institute of Technology, India

Abstract

The content based image retrieval (CBIR) of remotely sensed (RS) images is vital in the era of processing huge numbers of remotely sensed images. The paper implements a method for CBIR using HSV histograms for retrieving closely matching images from the database and a texture based strategy for forest fire detection. In texture based strategy Gray level co-occurrence Matrix (GLCM) has been used in combination with Feed Forward Neural Network to detect forest fire. The results presented in this paper were obtained through conducting experiments on IRS P6 AWiFS satellite images downloaded from Internet.

Keywords:

Remote Sensing, Histogram, Feature Extraction, Feature Vector, AWiFS

1. INTRODUCTION

Content Based Image Retrieval (CBIR) is a technique which uses visual features of image such as color, shape, texture, etc. for finding the closest image from the database. CBIR techniques provide a method to find images from large databases by using unique descriptors from trained images. A lot of research work has been completed in the past decade to design efficient image retrieval techniques from the image databases. As forest constitutes a large part of earth's renewable natural resources and plays a pivotal role in maintaining a near ideal environmental condition for life sustenance, there is a need for comprehensive international set of comparable data on forest fires for operational planning.

The second section gives a detailed literature survey of the existing methods applied in content based image retrieval. The third section gives detail about the experimental data used in this paper. The methodology for CBIR applied in this paper is discussed in the fourth section while the fifth section describes the experimental results. The final section draws conclusion.

2. LITERATURE SURVEY

CBIR is a hybrid research area that requires knowledge in both computer vision and database systems. It aims in retrieving the most similar image from a database for a given input image. CBIR based on multimedia technologies such as an image, video, audio, text files found application area in urban planning, disaster management, weather forecasting etc. The large numbers of images are challenges in computer systems. Few challenges are reported in the following literature. In a survey on different relevance feedback techniques in CBIR by Athira Mohanan, Sabitha Raju [2], they have discussed both feature based and subspace based relevance feedback technique and suggested that subspace learning method was more efficient than feature based methods. Another retrieval system by Rumelhart et al [12] proposed the classical feed forward neural networks which form the basis for today's deep learning neural networks. In a paper visual features of the contents in images were taken for retrieving the relevant images, where they considered local handcrafted, global handcrafted features and the Convolutional Neural Network (CNN) [7] and discussed that the proposed method was better than different retrieval schemes. In another work [1] the authors investigated three different quantization schemes and proposed for each one an efficient retrieval approach. More precisely, the uniform quantizer, the moment preserving quantizer and the distribution preserving quantizer were considered. The inherent properties of each quantizer were then exploited to design an efficient retrieval strategy to reduce the drop of retrieval performances resulting from the quantization effect. In order to provide high quality content-based search services over huge volume of image collections, multiple features were integrated. This might slow down the system because of high dimensionality occupied by the features space. So Artificial Neural Network (ANN) [4] was applied which considered only the selected features that offered high similarity with the query image. In order to minimize the labeling effect of the user in retrieving images from Support Vector Machine (SVM) framework, an Active Learning method to drive conventional Relevance feedback mechanism was proposed which evaluated three criteria that included diversity, density and uncertainty of the images [3].

In another paper [8] remote sensing datasets that included limited and plenty of labeled samples were taken and were trained using Deep Hashing Neural Network (DHNN). In order to increase the effectiveness of visual hashing without any explicit semantic label and also to achieve the target, a unified unsupervised framework was developed [5].

The application area of CBIR is multitudinous and so one can also provide data in encrypted form. For the privacy-preserving purposes sensitive images such as medical and personal images were to be encrypted before being outsourced and so CBIR technologies in plaintext domain became unusable. So a scheme that supported CBIR [10] over the encrypted images without revealing the sensitive information to the cloud server was done.

In [6] CBIR consisted of three methods which involved Geolocation-based image retrieval (GLBIR), Principal component analysis (PCA) and multiple region-based image retrieval. In the first method it dealt with identifying geo location of an image using visual attention based mechanism and its color layout descriptors were used for feature extraction from geo-location of query image. In the second method in order to filter images, unsupervised feature technique was integrated using principal component analysis (PCA). The visually similar images were clustered together and outliers were removed. In the third method, watershed Region of Interest (ROI) was employed for user interface. A variation in the retrieval accuracy was resolved by a method [9], which dealt with a three-layer framework that integrated the strengths of query expansion and fusion of holistic and local features to learn suitable query-dependent fusion weights to obtain better retrieval accuracy.

Haralick et al. [11] introduced Gray-Level Co-occurrence Matrix (GLCM) and 14 texture measures for extracting texture features of gray-level images.

3. EXPERIMENTAL DATA

In this paper, the dataset taken is IRS (Indian Remote-Sensing Satellite) P6 AWiFS satellite images consisting of burnt area caused by forest fire. The main objectives of the IRS-P6 satellite (ResourceSat-1) are to provide continued remote sensing data services on an operational basis for integrated land and water resources management. The dataset taken for this work contains the Part of India during the year 2012-13 [https://nrsc.gov.in/hackathon2018].

In this duration forest fire took place. The proposed work addresses the CBIR problem given by National Remote Sensing Centre in the Hackathon 2018 link. Data provided is raster data in .tiff format. There is a separate .tiff file for each band. The data is received from Advanced Wide Field Sensor (AWiFS) which operates in three spectral bands in Very Near Infra-red (VNIR) and one spectral band in Short Wave Infrared (SWIR) with 56m spatial resolution and a combined swath of 730km achieved through two AWiFS cameras.

Table.1. AWiFS Specification and Spectral Resolution

AWiFS Specification	
Spatial Resolution	56m
Band	**Wavelength (μm)**
Band B_2 (VIS)	0.52 to 0.59
Band B_3 (VIS)	0.62 to 0.68
Band B_4 (NIR)	0.77 to 0.86

4. METHODOLOGY

The paper comprises of two phases. In the first phase, in order to perform CBIR of RS images HSV histogram and minimum distance classifier are used. In the second phase, texture based methodology uses Gray level co-occurrence matrix (GLCM) and Feed forward neural network for forest fire detection.

4.1 HISTOGRAM BASED METHODOLOGY

Histogram-based data analysis is one of the most popular solutions for many problems related to image processing. The methodology is represented in the flow chart in Fig.1. In general, color channel histogram preserves the color information in an image. The distances between the color histogram of query image and the color histograms of database images are calculated. The distance values are then sorted and the resultant images are obtained from the database based on the minimum distance values.

4.1.1 HSV Histogram Feature Extraction:

The band combination of 2, 3 and 4 bands of RS image can be performed in GIS software by projecting the contents of each band in Red, Green and Blue channels respectively. The query image is read and then to preserve the aspect ratio of the output image, the image is resized. RGB color space describes colors in terms of the red, green, and blue additive primaries. HSV color space describes colors in terms of the Hue, Saturation and Value components. RGB to HSV conversion is done to represent the image in terms of two color components namely ‘Hue’ and ‘Saturation’ and one intensity value component namely ‘Values’. The conversion helps in retaining the color information inherent in the image. In order to extract features, the histogram for hue, saturation and value channels of the images are found.

4.1.2 Distance Measurement:

In image analysis, the distance metric measures the distances between the feature vector of the query image and the feature vectors of training images. The Euclidean distance is the straight-line distance between the feature vectors of two images. So the Euclidean distance is measured between the HSV histograms of training images in the database and the HSV histogram of query image.

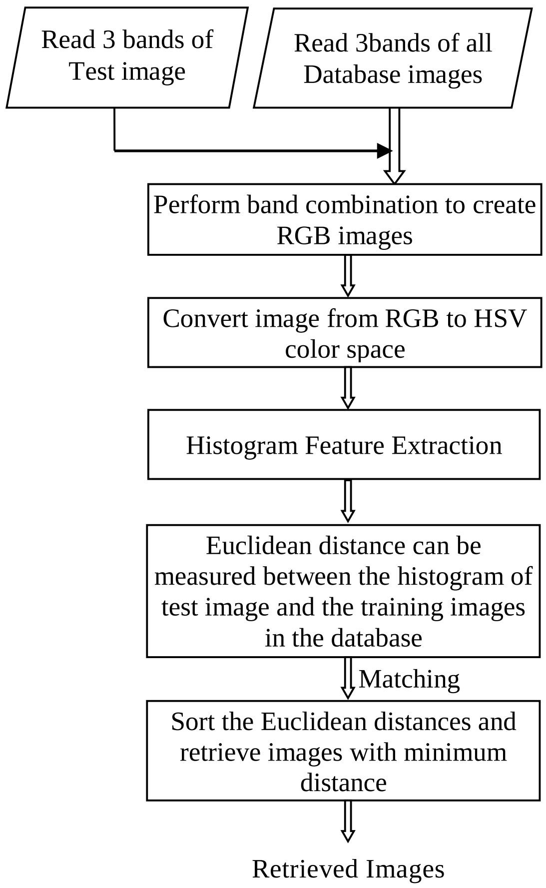


Fig.1. Outline of the work

4.1.3 Matching:

Finally, the images in the database are sorted in increasing order based on the distance values calculated with the query image. The images most relevant to the query image are retrieved based on minimum distance.

4.1.4 Implementation:

The pseudo-code for the methodology adapted is shown below.

```
% Read the test image
img1=imread('test image.jpg')
% Convert image to HSI color space
im1=rgb2hsv(img1);
h=im1(:,:,1);
s=im1(:,:,2);
v=im1(:,:,3);
% Form RGB histograms for the test image
[hCount,hValues]=hist(h(:),18);
[sCount,sValues]=hist(s(:),3);
[vCount,vValues]=hist(v(:),3);
his1=[hCount sCount vCount];
his=his1';
% Read all the training images
% Let the file names of training images be 1.jpg, 2.jpg,..., 10.jpg
for i=1:10
filename=strcat(num2str(i),'.jpg');
im1=imread(filename);
% convert to HSI color space and separate the bands
im1=rgb2hsv(im1);
h=im1(:,:,1);
s=im1(:,:,2);
v=im1(:,:,3);
% Form RGB histogram of training image
[hCount,hValues]=hist(h(:),18);
[sCount,sValues]=hist(s(:),3);
[vCount,vValues]=hist(v(:),3);
thist=[hCount sCount vCount];
hi(:,i)=thist;
end
% Find the Euclidean distance between the test sample and training samples
for(i=1:10)
d=(his(:)-hi(:,i))^2;
e(i)=sqrt(sum(d(:)));
end
%Sort the distances
[sv sp]=sort(e);
% Display the closely matching three images from the database
for i=1:3
temp=strcat(sp(i),'.jpg');
im=imread(char(temp));
figure(i)
imshow(uint8(im));
end
```

4.2 TEXTURE BASED METHODOLOGY

4.2.1 GLCM Feature Extraction:

The steps followed in creating a symmetrical normalized GLCM [11] for a gray level image are as follows. A framework matrix of 256×256 is created because the gray levels range from 0 to 255. The spatial relation is decided between the reference and neighbour pixel. The number of occurrences is counted and the framework matrix is filled. For example, GLCM has been calculated for the 4×4 matrix given below in Fig.2 keeping 'zero right one down' (0R1D) spatial relation. The matrix is added to its transpose to make it symmetrical and turning the entries into probabilities creates a normalized matrix.

$$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ \end{array} \qquad\qquad \begin{array}{ccccc} & 1 & 2 & 3 & 4 \end{array}$$
$$\begin{bmatrix} 4 & 4 & 1 & 1 \\ 3 & 3 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 2 & 2 & 3 & 3 \end{bmatrix} \xrightarrow{GLCM} \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array}\begin{bmatrix} 2 & 3 & 0 & 0 \\ 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix}$$

Fig.2. GLCM Calculation

Haralick et al. [11] proposed 14 different texture measures that can be computed from GLCM. They are grouped into 3 categories.

1. Measures related to contrast that use weights related to the distance from the GLCM diagonal.
2. Measures related to orderliness.
3. Measures using descriptive statistics of the GLCM.

These measures together give good pattern discrimination of texture based images. The contrast, correlation, energy and homogeneity are some of the 14 texture measures related to contrast, orderliness and statistics and are defined as below.

$$Contrast = \sum_{i,j=1}^{256} (i-j)^2 GLCM(i,j) \tag{1}$$

$$Homogeneity = \sum_{i=1}^{256}\sum_{j=1}^{256} \frac{GLCM(i,j)}{1+|i-j|} \tag{2}$$

$$Energy = \sqrt{\sum_{i,j=1}^{256} \left(GLCM(i,j)\right)^2} \tag{3}$$

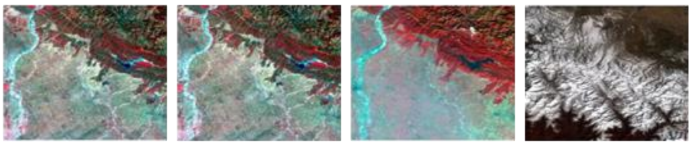
$$Mean(\mu_i) = \sum_{i,j=1}^{256} i\left(GLCM(i,j)\right) \tag{4}$$

$$SD(\sigma_i) = \sqrt{\sum_{i,j=1}^{256} GLCM(i,j)(i-\mu_i)^2} \tag{5}$$

$$Correlation = \sum_{i,j=1}^{256} GLCM(i,j)\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{(\sigma_i^2)(\sigma_j^2)}} \tag{6}$$

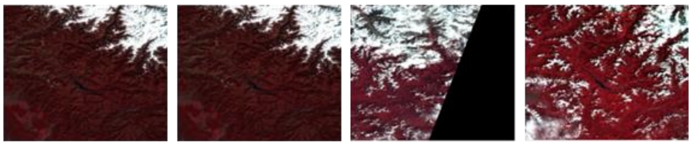
4.2.2 Feed Forward Neural Network:

A feed forward neural network works without back propagation and so it does not have any cycle. Multiple Sigmoid neurons combine to form the feed forward neural network.



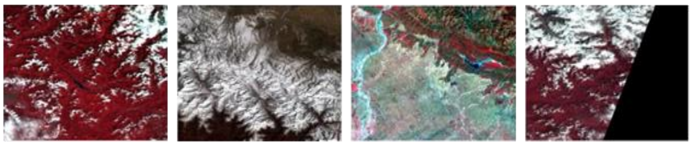
(a) (b) (c) (d)

Fig.3. Image 1: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



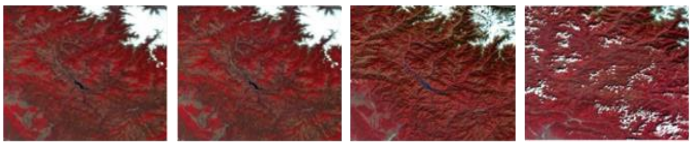
(a) (b) (c) (d)

Fig.4. Image 2: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



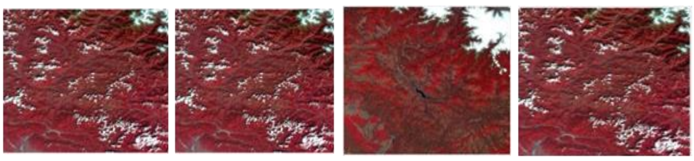
(a) (b) (c) (d)

Fig.5. Image 3: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



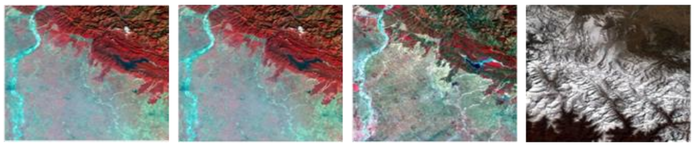
(a) (b) (c) (d)

Fig.6. Image 4: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



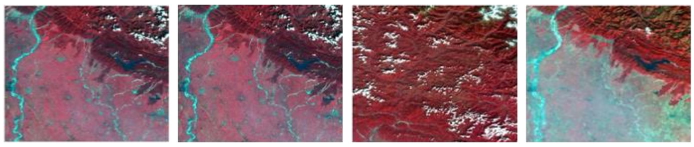
(a) (b) (c) (d)

Fig.7. Image 5: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



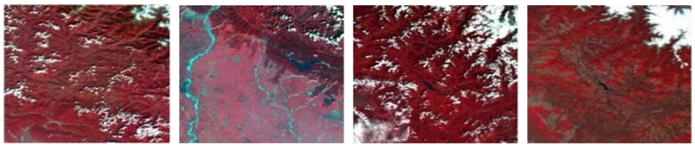
(a) (b) (c) (d)

Fig.8. Image 6: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



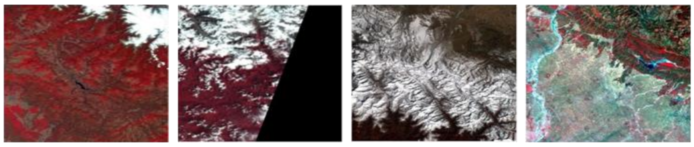
(a) (b) (c) (d)

Fig.9. Image 7: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



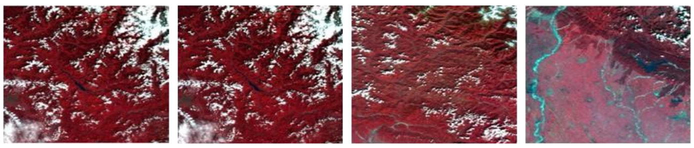
(a) (b) (c) (d)

Fig.10. Image 8: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



(a) (b) (c) (d)

Fig.11. Image 9: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4



(a) (b) (c) (d)

Fig.12. Image 10: In each row, query image is shown in column 1 and the closely matching three retrieved images are shown in columns 2, 3 and 4

The flow of information takes place in the forward direction, as input is used to calculate some intermediate function in the hidden layer which in turn is used to calculate output. The layers between the input layer and output layers are known as hidden layers, as the training data does not show the desired output for these layers. A network can contain any number of hidden layers with any number of hidden units. A unit basically resembles a neuron which takes input from units of previous layers and computes its own activation value. The hidden layers present in between the input and output layers are used to handle the complex non-linearly separable relations between the input and output.

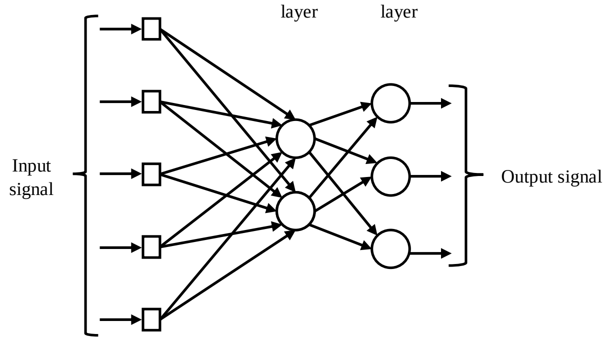


Fig.13. Sample Feed Forward Neural Network

4.2.3 Classification using Feed Forward Neural Network:

The Feed forward neural network has two phases namely training and testing phases. In the training phase, feed forward neural network is trained with the training images and in the testing phase, a query image is read and the output stating whether the query image has forest fire or not is returned based on the prior training. In the set of training images considered for training the feed forward neural network, one half of the training images have forest fire and the other half does not have forest fire. In the training set matrix, each row consists of the Contrast, Correlation, Energy and Homogeneity texture features of each training image and its corresponding class label. Each image has three bands and for each band four GLCM features are computed. So each image has 12 features. The feed forward neural network has 12 input neurons in the input layer and the output layer has two neurons that return whether the image has forest fire or not. The neurons in the hidden layers have activation functions that try to map the inputs to output.

4.2.4 Implementation:

```
% Read all the training images
% Let the file names of RGB training images be 1.jpg, 2.jpg,..., 40.jpg
% Find GLCM texture measures
k=zeros(1,12);
for i=1:40
s=strcat(num2str(i),'.jpg');
im=imread(s);
h=glcmcalculation(im);
temp1=
[h(1).Contrast h(1).Correlation h(1).Energy h(1).Homogeneity h(2).Contrast h(2).Correlation h(2).Energy h(2).Homogeneity h(3).Contrast h(3).Correlation h(3).Energy h(3).Homogeneity];
k=cat(1,k,temp1);
end
net=feedforwardnet(10);
in(:,:)=k(2:end,:);
% Let tar of size (40, 1) contains the % target class labels of 40 training % samples. Use 1 for presence of forest
% fire and 0 for its absence.
% Train Feed forward neural network with % the training set
net=train(net,in',tar');
view(net)
cl=net(in');
perf = perform(net,tar',round(cl));
% testing phase
% Read the RGB test image
img1=imread('test image.jpg')
% Convert image to HSV color space
im1=rgb2hsv(img1);
% Find GLCM texture measures
t=glcm(im1);
temp2=
[t(1).Contrast t(1).Correlation t(1).Energy t(1).Homogeneity t(2).Contrast t(2).Correlation t(2).Energy t(2).Homogeneity t(3).Contrast t(3).Correlation t(3).Energy t(3).Homogeneity];
% Test the Feed forward neural network % with the feature vector of test image
class=net(temp2');
if(class>=.5)
  disp('Forest fire detected');
else
  disp('No forest fire');
end
% Function that calculates GLCM texture % measures
function [g]=glcm(temp)
for i=1:3
temp1(:,:)=temp(:,:,i);
GLCM=graycomatrix(temp1, 'Numlevels',256, 'GrayLimits',[]);
g(i,:)=graycoprops(GLCM,{'Contrast', 'Correlation', 'Energy', 'Homogeneity'});
end
end function
```

5. EXPERIMENTAL RESULTS

The experimental results consist of the IRS P6 RS images taken in different months of the year 2012-2013. In the first phase of work on CBIR, for every month a query image has been taken and the HSV histogram feature vector is extracted. The distance values between the histogram feature vector of the query image and the histogram feature vectors of the training images present in the database are measured and sorted. The Fig.3(a) to Fig.12(a) represent the query images chosen from the database and Fig.3(b), Fig.3(c) and Fig.3(d) to Fig.12(b), Fig.12(c) and Fig.12(d) represent the retrieved images with minimal distances. From the results obtained in the experiments, it is vividly seen that the algorithm retrieves the most relevant images accurately. In the second phase of work on forest fire detection, four texture features

(Contrast, Correlation, Energy and Homogeneity) for each band of RGB image are extracted. So the size of the input layer is fixed to 12. The size of the output layer is 2 since the presence or absence of forest fire in the RS image is detected while the size of the hidden layer is fixed to 10. The Levenberg-Marquardt optimization training function is chosen to map inputs to output. The maximum number of epochs fixed is 100. The total number of training images used in this work is 40. The size of each training image is (256×256). The trained Feed forward neural network is shown in Fig.14. A sample input query image in which the algorithm detected forest fire and a sample query image in which the algorithm detected no forest fire are shown in Fig.15 and Fig.16.

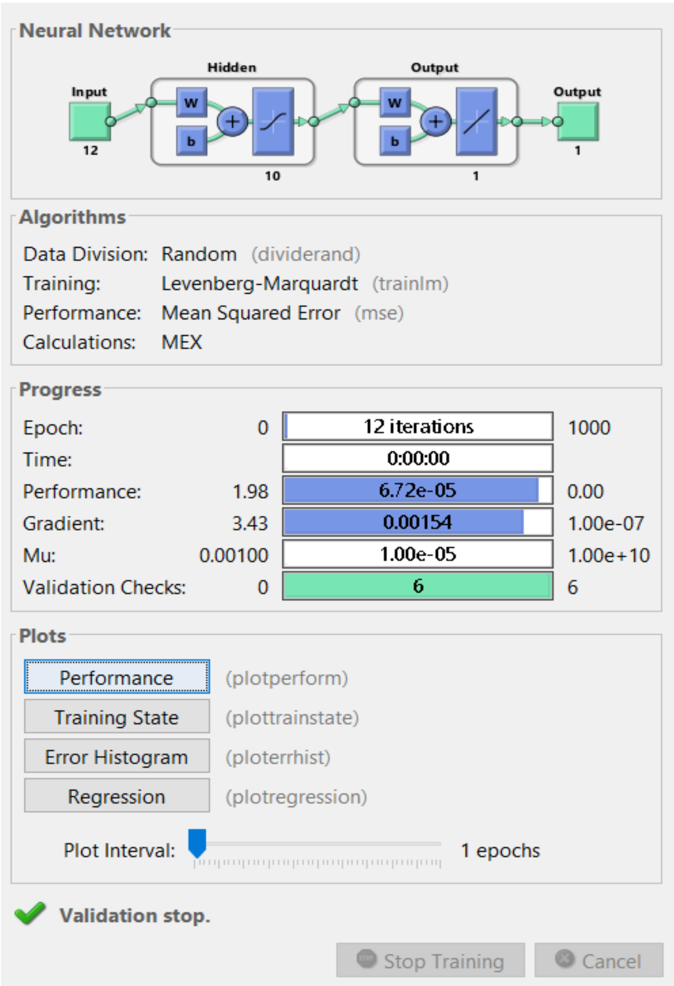


Fig.14. Training Feed forward neural network

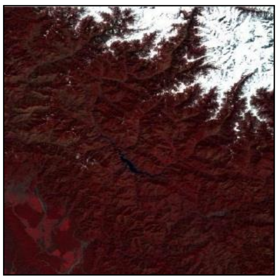


Fig.15 Query image in which the algorithm detected forest fire

Fig.16. Query image in which the algorithm detected No Forest Fire

6. CONCLUSIONS

In this paper, initially HSV histogram feature extraction and Euclidean distance based comparison have been used for image retrieval. Furthermore, forest fire in a satellite image has been effectively detected using GLCM and feed forward neural network. The adapted methodologies are computationally simple and take only minimum time for processing.

REFERENCES

[1] Amani Chaker, Mounir Kaaniche, Amel Benazza Benyahi and Marc Antonini, "Efficient Transform-based Texture Image Retrieval Techniques under Quantization Effects", *Multimedia Tools and Applications*, Vol. 77, No. 1, pp. 1-25, 2018.

[2] Athira Mohanan and Sabitha Raju, "A Survey on Different Relevance Feedback Techniques in Content Based Image Retrieval", *International Research Journal of Engineering and Technology*, Vol. 4, No. 2, pp. 582-585, 2017.

[3] B. Demir and A. Lorenzo, "A Novel Active Learning Method in Relevance Feedback for Content-Based Remote Sensing Image Retrieval", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 53, No. 5, pp. 2323-2334, 2015.

[4] M.E. ElAlami, "A New Matching Strategy for Content based Image Retrieval System", *Applied Soft Computing*, Vol. 14, No. 3, pp. 407-418, 2014.

[5] Lei Zhu, Jialie Shen, Liang Xie and Zhiyong Cheng, "Unsupervised Visual Hashing with Semantic Assistant for Content-Based Image Retrieval", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 29, No. 2, pp. 472-486, 2017.

[6] Muhammad Hammad Memon, Jian Ping Li, Imran Memon and Qasim Ali Arain, "GEO Matching Regions: Multiple Regions of Interests using Content based Image Retrieval based on Relative Locations", *Multimedia Tools and Applications*, Vol. 76, No. 14, pp. 15377-15411, 2017.

[7] Paolo Napoletano, "Visual Descriptors for Content-Based Retrieval of Remote Sensing Images", *International Journal of Remote Sensing*, Vol. 39, No. 5, pp. 1343-1376, 2017.

[8] Yansheng Li, Yongjun Zhang, Xin Huang, Hu Zhu and Jiayi Ma, "Large-Scale Remote Sensing Image Retrieval by Deep Hashing Neural Networks", *IEEE Transaction on Geoscience and Remote Sensing*, Vol. 56, No. 2, pp. 950-965, 2018.

[9] Yuebin Wang, Liqiang Zhang, Xiaohua Tong, Liang Zhang, Zhenxin Zhang, Hao Liu, Xiaoyue Xing and P. Takis Mathiopoulos, "A Three-Layered Graph-Based Learning Approach for Remote Sensing Image Retrieval", *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 54, No. 10, pp. 6020-6034, 2016.

[10] Zhihua Xia, Neal N. Xiong, Athanasios V. Vasilakos and Xingming Sun, "EPCBIR: An Efficient and Privacy-Preserving Content-Based Image Retrieval Scheme in Cloud Computing", *Information Sciences*, Vol. 387, pp. 195-204, 2017.

[11] R.M. Haralick and K. Shanmugam, "Textural Features for Image Classification", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 3, No. 6, pp. 610-621, 1973.

[12] D.E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning Representations by Back-Propagating Errors", *Nature*, Vol. 323, pp. 533-536, 1986.