NPFAM: Non-Proliferation Fuzzy ARTMAP for Image Classification in Content Based Image Retrieval

Anitha K[1] and Chilambuchelvan A[1]
[1] R M K Engineering College, Anna University
Chennai - India
[e-mail: anithak.rameshr@gmail.com]
[e-mail: chill97@gmail.com]
*Corresponding author: K.Anitha

Received August 4, 2014; revised October 24, 2014; revised December 21, 2015; accepted June 15, 2015; published July 31, 2015

Abstract

A Content-based Image Retrieval (CBIR) system employs visual features rather than manual annotation of images. The selection of optimal features used in classification of images plays a key role in its performance. Category proliferation problem has a huge impact on performance of systems using Fuzzy Artmap (FAM) classifier. The proposed CBIR system uses a modified version of FAM called Non-Proliferation Fuzzy Artmap (NPFAM). This is developed by introducing significant changes in the learning process and the modified algorithm is evaluated by extensive experiments. Results have proved that NPFAM classifier generates a more compact rule set and performs better than FAM classifier. Accordingly, the CBIR system with NPFAM classifier yields good retrieval.

Keywords: Fuzzy ARTMAP, Image classifier, Category proliferation, Image retrieval

1. Introduction

Content Based Image Retrieval (CBIR) system effectively searches image database and retrieves particular images relevant to query image submitted by user. CBIR system focuses on visual content of images rather than manual annotation. Images are thus indexed by their individual visual features [1-2] such as color, texture and shape. An extensive variety of image descriptors are provided by MPEG7 [3] to extract these visual features. Selection of descriptors greatly affects performance of the CBIR system. From literature it is evident that extracting fixed set of features from images fails to give satisfactory results [4-5]. To overcome the problems in selecting feature sets, a CBIR system that adopts different feature sets for different classes was proposed in [6]. The work in [7] provides convincing retrieval results by using combined shape and color features with relevance feedback.

Retrieval efficiency can be improved by various methods [8-11]. Statistical model and radial basis function neural network has been presented in [8] for color image retrieval. In [9] a hybrid meta-heuristic swarm intelligence-based search method is used for image retrieval. Problems in conventional distance and classifier based retrieval approaches was addressed in [10] using a fuzzy class membership. Retrieval accuracy was improved by user oriented image retrieval system based on interactive genetic algorithm [11]. Fuzzy algorithms are extensively used for classification in pattern recognition applications like image retrieval. Adaptive Resonance Theory (ART) was developed by Grossberg [12]. This architecture possesses many desirable properties that can solve arbitrarily complex classification problems. Fuzzy Artmap (FAM) [13] which better classifies input patterns can be implemented as the matching machine of CBIR system. Extension and modification of FAM neural classifier proposed for face recognition systems in [14-15] has made effective feature classification.

Limitation faced by FAM is category proliferation problem reported in [16]. This refers to the case where FAM classifier creates unnecessarily large IF-THEN rules while dealing with large and noisy data sets. Distributed Artmap that combines computational advantages of Multi Layer Perceptron (MLP) and Adaptive Resonance Theory (ART) systems in real time neural network for supervised learning is presented in [17] and its performance was verified with a circle in square problem. A hybrid neural network classifier of FAM and Dynamic Decay Adjustment (DDA) algorithm is presented in [18]. The algorithm proves to decrease misclassification rate. Evidence in [19] explores that Boosted Artmap as a viable learning technique over FAM. Safe µArtmap described in [20] limits the growth of a category in response to a single pattern, so that large hyperboxes are not created. In [21], number of modifications is implemented in FAM to reduce the category proliferation problem caused due to overlapping classes. This paper proposes a modified FAM known as Non Proliferation Fuzzy Artmap (NPFAM) which focuses on avoiding proliferation problem. Once low level features are extracted using MPEG7 descriptors, they are fused into NPFAM classifier for training process. During retrieval the classifier matches a class for given query image. After system identifies the class of query image, similarity between images in the class and query image is calculated using Euclidean distance similarity metric. Images in the class with higher similarity are presented to user.

The organization of this paper is as follows: Section 2 presents details of descriptors used in feature set. Section 3 describes FAM classifier. Section 4 gives details about NPFAM classifier. Section 5 provides experimental data sets used and performance comparison between FAM and NPFAM. Finally Section 6 gives conclusion and future work.

2. Feature Space and Formation of Feature Space

Two prominent steps involved in CBIR systems are indexing and searching. Indexing computes essential features that describe the image to its best and stores them in database as feature space. Searching computes feature vectors for query image, compares them with feature vectors of images in database and images which are most similar to query image are presented to the user. This section provides brief details of MPEG7 visual content descriptors included in feature space.

2.1 Visual Descriptors

In literature, a variety of descriptors has been discussed for CBIR systems. MPEG7 have formed four types of visual descriptors namely i) Movement descriptor ii) Texture descriptor iii) Color descriptor and iv) Shape descriptor. Detailed studies of these descriptors are presented in [22]. Descriptors used for indexing in the proposed system are briefed below.

2.1.1 Color Descriptor

Color descriptors originating from histogram analysis have played a central role in the development of visual descriptors in MPEG7.

Color Layout Descriptor (CLD) is a spatial distribution of colors in compressed form of a segmented or complete image. Compactness of descriptor provides better retrieval results and supports sketch based retrieval. Representative colors on a sliding 8X8 grid are encoded by DCT and used in CLD. YCrCb color space is adopted and contributes 192 coefficients (64Y, 64Cr, 64Cb) in the feature set which is used to index images.

Scalable Color Descriptor (SCD) defined in HSV color space with a uniform quantization to 256 bins, addresses interoperability issue available in generic color histogram descriptor. SCD is a color histogram encoded by Haar transform. Full interoperability between different resolutions of color representation is achieved by ranging 16 bits/histogram to 1000 bits/histogram. From experiments it was found that 64 bits/histogram yield good retrieval results. SCD has 64 coefficients in the feature set representing the image.

Color Structure Descriptor (CSD) uses HMMD color space to describe local color structure of an image in the feature set. By scanning the image using an 8X8 structure, presence of a particular color in the structure is counted. A color histogram is constructed such that it provides the count of a particular color in an image. CSD contributes 32 coefficients in the feature set.

2.1.2 Shape Descriptor

This feature provides a strong visual hint for similarity matching and is not influenced by scaling, rotation and translation. Shape transformation can be 2D or 3D. In general 2D shape description can be region or contour-based.

Region-based descriptor-ART is a moment invariant descriptor constructed by a complex angular radial transformation. All pixels comprising the shape bounded within a frame are used. This feature has the ability to describe any shape such as connected or disjoint regions. It can sustain minimal distortions along the object's boundary. The descriptor is represented with 36 coefficients in the feature set.

2.1.3 Texture Based Descriptors

Texture Browsing Descriptor (TBD), Homogenous Texture Descriptor (HTD) and Local Edge Histogram Descriptor (EHD) are the low level features that can represent the texture of an

image. This descriptor enhances image search and retrieval process. Dissemination of texture in an image is described by Homogenous Texture Descriptor (HTD). It improves classification accuracy of objects in image retrieval. HTD is composed of 62 coefficients with first two as mean, standard deviation and remaining are energy and energy deviation of channels in frequency domain of Gabor filter responses. Feature vector of size 60 (30 means and 30 variances) for 5 scales each with 6 points of references, are obtained, scaled to analog [0-1] and placed into global feature vector.

Edge Histogram Descriptor (EHD) captures spatial dissemination of edges. An image is partitioned into sixteen equal isolated blocks covering the complete image. Edge information is then estimated for each block in five edge categories: 0^o, 45^o, 90^o, 135^o and non-directional edge, each with one bin of local histogram. Therefore EHD gives 80 (16 non overlapped partitioned image blocks X 5 bins for 5 categories) of its coefficients to global feature vector. Thus global vector of size 464 is created to symbolize each image in the data set.

3. Fuzzy ARTMAP

Fuzzy ARTMAP architecture [13] derived from ART, has been widely adopted in literature for classification. It performs faster and expedites stable learning. The architecture includes two unsupervised fuzzy ART modules. These modules partitions input and output spaces. Fuzzy ART modules and Fuzzy ARTMAP are briefed in this section.

3.1 Fuzzy ART

Fuzzy ART is a binary ARTMAP that performs unsupervised learning implemented in analog domain using fuzzy and ($\wedge$) operator. The architecture comprises below fields.

F_0: Input field node used to store input vectors denoted by $I = (I_1, I_2, ..., I_M)$ where $I_i \in [0,1]$ and M is number of nodes.

F_1: Match field with activity vectors denoted by $x = (x_1, x_2, ..., x_M)$

F_2: Output field with category vectors $y = (y_1, y_2, ..., y_N)$ where N is number of nodes.

Accumulation of input patterns is modeled by weights. Each input pattern has a stronger association with an output category. Thus each individual node j of F_2 is associated with a weight vector $w_j = (w_{j1}, w_{j2}, ..., w_{jM})$. Weight w_j represents the degree to which a particular feature is present in the image a such that it is coded by j^{th} category. Initially the weights of output nodes are assigned to one and will not be selected by input pattern. During training an input pattern of known class is presented to the system. one of the output nodes j is selected by the pattern. The selected node is considered to represent the class of input pattern. Once a node in F_2 layer gets selected, it is considered as committed to the input and weights are adjusted. Weights will always decrease and remains stable after convergence. Each output node is called as category and one or more categories may represent one class.

Choice Function: Consider N previously committed nodes in output layer. If an input pattern is presented, n nodes out of N committed nodes (n=1 or 2 or….or N) with varying degrees can be selected by the pattern. From n selected nodes at most one node (j) in F_2 can become active, by taking on the input pattern (I). Output node (j) that can become active for input (I) is selected by evaluating choice (Target) function as given by (1).

$$T_j(I) = \frac{|I \wedge w_j|}{\alpha + |w_j|} \tag{1}$$

where $\alpha \cong 0$ and $T_j(I)$ is called activation value that determines the active category among nodes j and k committed to an input pattern (I). Node that evaluates to the highest activation value becomes active node as given by (2).

$$\text{Active node} = T_J(I) = \max_j \{T_j(I)\} : j = 1, 2, ..., N \tag{2}$$

$T_j(I)$ determines strength of match between input pattern presented at an instant and weights of the j^{th} output node. The ratio $\frac{|I \wedge w_j|}{|w_j|}$ gives fuzzy subset hood of w_j with respect to I. If any w_j available is a fuzzy subset of I then $|I \wedge w_j| = |w_j|$ and $\frac{|I \wedge w_j|}{|w_j|} = \frac{|w_j|}{|w_j|} = 1$. Therefore, $T_j(I) > T_k(I)$ for $k \neq j$.

Resonance: Activity vector x in match field F_1 is formed as below.

$$x = \begin{cases} I & \text{If } F_2 \text{ is inactive} \\ I \wedge w_j & \text{If the } j^{th} \text{ node in } F_2 \text{ is active} \end{cases} \tag{3}$$

After a category J in output field matches to input pattern and becomes active, match field determines the match between input pattern and active category using (4).

$$\frac{I \wedge w_j}{|I|} \tag{4}$$

If activity vector x is similar to input vector then (4) will pass the vigilance test $\frac{I \wedge w_J}{|I|} \geq \rho$ where $\rho \in [0,1]$. If vigilance criteria is satisfied, network is said to be in resonance and the j^{th} node in F_2 is good enough to encode input pattern (I). In addition, for a node to establish resonance node j should also represent the same class as that of input pattern.

Learning: When an output node, that matches the input pattern is identified resonance happens, the weight vector w_j is updated by (5)

$$w_j^{new} = \beta(I \wedge w_j^{old}) + (1 - \beta) w_j^{old} \tag{5}$$

where β is learning rate parameter. $\beta \in [0,1]$ and $\beta = 1$ for fast learning.

Testing: Input pattern belongs to class corresponding to category which has high activation value. After training the system, using randomly selected input patterns the system is ready for classifying unknown images.

3.2 Fuzzy ARTMAP

Fuzzy ARTMAP [13], architecture performs supervised learning. Architecture shown in **Fig. 1**, has two ART modules, ART^a and ART^b. These modules cluster patterns from input space and output space into categories. During supervised learning, provided a set of input patterns $\{a, b\}$, ART^a receives a stream of input patterns $\{a\}$ and ART^b receives a stream of patterns $\{b\}$. These two modules are linked by an associative learning network and an internal controller. Inter art module F^{ab} (map field) links ART^a and ART^b.

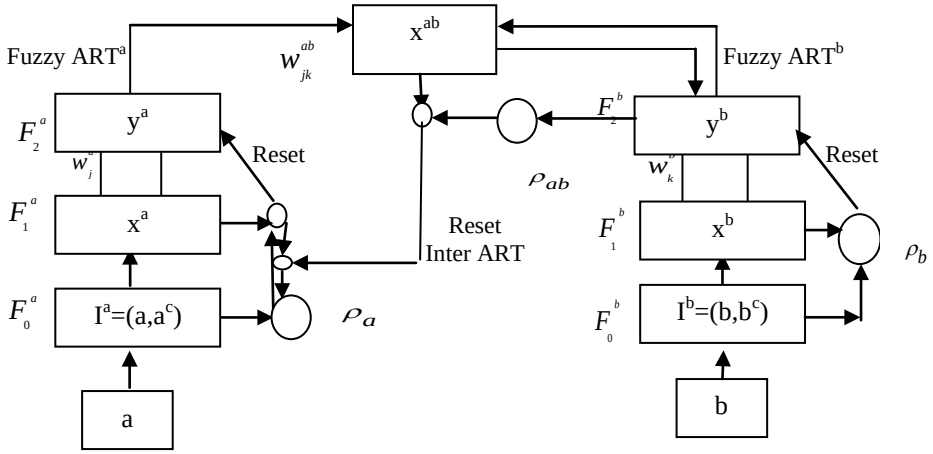


Fig. 1. Fuzzy ARTMAP Architecture

Map field gets triggered, when one of the ART^a or ART^b categories are active. Mapping field contains the association between categories predicted by ART^a and ART^b for an input pattern. If prediction made by module a is disconfirmed at ART^b, map field initiates match tracking process. Match tracking increments vigilance ρ^a and a new search is triggered.

The process converges, if search results in any of the ART^a category predicted matches the predicted category in ART^b or a previously uncommitted ART^a category node. This process ensures that the category that resonates matches to the best with input pattern. FAM system is same as Fuzzy ART. In addition; match ensuring mechanism is included.

Inputs to two FAM modules are in complement code $I^a = (a, a^c)$ and $I^b = (b, b^c)$ where vector $a = (a_1, a_2, ..., a_M)$ is input pattern of size M and $a^c = (1-a_1, 1-a_2, ..., 1-a_M)$ expresses absence of each feature in a. Input vectors I^a and I^b will be of size 2M. The map field $x^{ab} = (x_1^{ab}, x_2^{ab}, ..., x_N^{ab})$ denotes output vector of F^{ab} and weight vector of j^{th} node to F^{ab} is denoted by $w_j^{ab} = (w_{j1}^{ab}, w_{j2}^{ab}, ...w_{jN}^{ab})$. Between input presentations all activity vectors are set to zero.

3.2.1 Map field Establishment

Input to the mapping field F^{ab} is from both or any one of the category fields in ART^a and ART^b. That is its activity vector x^{ab} is constituted by F_2^a and F_2^b as given by (6).

$$x^{ab} \begin{cases} y^b \wedge w_J^{ab} & \text{If } J^{th} \ F_2^a \text{ node and } F_2^b \text{ node are active.} \\ w_J^{ab} & \text{If } J^{th} \ F_2^a \text{ node and } F_2^b \text{ node are inactive.} \\ y^b & \text{If } F_2^a \text{ is inactive and } F_2^b \text{ is active.} \\ 0 & \text{If } F_2^a \text{ and } F_2^b \text{ are inactive} \end{cases} \tag{6}$$

If for a given input pattern, J^{th} category in F_2^a wins and becomes active, it sends information to map field F^{ab} through w_j^{ab}. Then F^{ab} is active only if F_2^a predicts same class as F_2^b. If y^b fails to confirm prediction made by w_j^{ab}, then $x^{ab} = 0$ and match tracking is initiated.

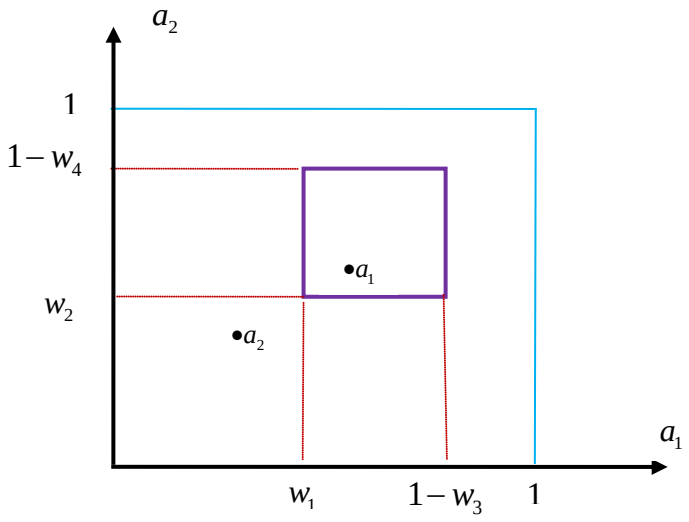


Fig. 2. Representation of hyperbox with size R_1 formed by weights $w=(w_1,w_2,w_3,w_4)$ in two dimension input space and input pattern a_1 inside the box and a_2 outside the box

3.2.2. Match Tracking

When the input is first presented to the network the vigilance parameter ρ^a is set to its baseline value $\overline{\rho^a}(0<\rho^a<1)$. Matching field vigilance parameter ρ^{ab} ensures the matching between categories in ART^a and ART^b. if $\frac{|x^{ab}|}{|y^b|}>\rho^{ab}$ then a best match exists. If a matching error occurs i.e. $\frac{|x^{ab}|}{|y^b|}<\rho^{ab}$, match tracking increments $\rho^a>\frac{|I^a\wedge w_J^a|}{|I|}$ and search is initiated for finding new F_2^a node. The process continues till a correct match for I^a is identified by ART^a. Otherwise a new category is created in ART^a and committed to I^a.

3.2.3 Category Proliferation

FAM suffers from category proliferation problem. Exclusive advantage of FAM is that its weights can be easily converted into IF-THEN rules. Number of categories created during training FAM is usually large. This causes an intractable bunch of IF-THEN rules due to class mismatches and creation of new categories which leads to category proliferation problem. This section briefs about how match tracking process leads to creation of new categories.

FAM category (j) can be viewed as hyperboxes R_j with corners decided by weights w_j. Weight vector (w_{ji}^{ab}) of j^{th} node in inter art module and its complement $(1-w_{j,m+i})$ gives the minimum and maximum values of i^{th} component among patterns in input vector that has selected j^{th} category. So category size R_j can be defined by (7).

$$|R_j|=M-|w_j|=\sum_{i=1}^{M}l_{ji} \tag{7}$$

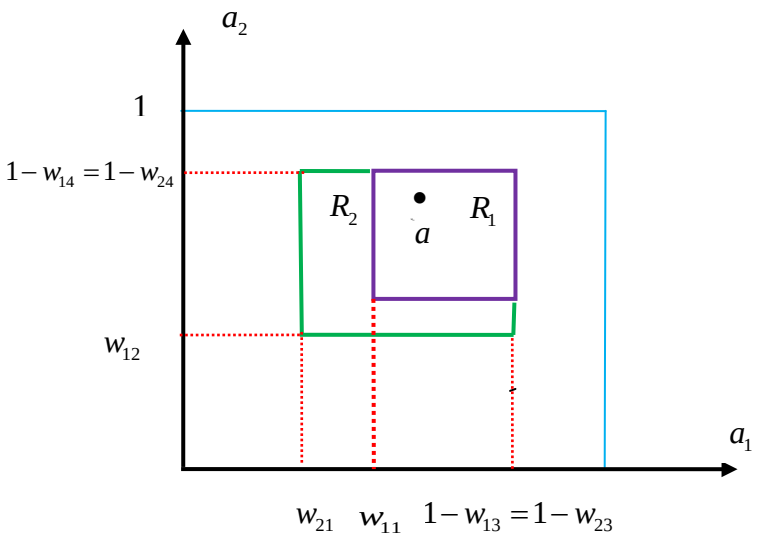


Fig. 3. Illustration to prove if input pattern a contained in one or more hyperboxes R_j, the smaller box will have higher activation value. $w_1=(w_{11},w_{12},1-w_{13},1-w_{14})$; $w_2=(w_{21},w_{22},1-w_{23},1-w_{24})$;

$$T_1=\frac{\min(a_1,w_{11})+\min(a_2,w_{21}+\min(1-a_1,w_{13})+\min(1-a_2,w_{14})}{\alpha+w_{11}+w_{12}+(1-w_{13})+(1-w_{14})}=\frac{w_{11}+w_{12}+(1-w_{13})+(1-w_{14})}{\alpha+w_{11}+w_{12}+(1-w_{13})+(1-w_{14})}$$

$=\frac{|w_1|}{\alpha+|w_1|}$; Similarly, $T_2=\frac{w_{21}+w_{22}+(1-w_{23})+(1-w_{24})}{\alpha+w_{21}+w_{22}+(1-w_{23})+(1-w_{24})}=\frac{|w_2|}{\alpha+|w_2|}$; Therefore $T_1<T_2$, Since

$(w_{11}+w_{12})<(w_{21}+w_{22})$ and $(1-w_{13})+(1-w_{14})=(1-w_{23})+(1-w_{24})$

where $M=|I|=a_1+a_2+(1-a_1)+(1-a_2)$ provides the size of attributes in input pattern and l_{ji} gives i^{th} component of patterns learnt by j^{th} category. Thus a pattern is learned by a category if pattern lies inside the box or the box enlarges to accommodate the pattern. The box is decided by choice function (1). Vigilance parameter ρ^a controls the size of the box. Hyperbox with weights w will be selected by an input pattern I only if $\rho^a<\frac{|I^a\wedge w_j^a|}{|I|}$. If the hyperbox passes the test, the weights of the hyperbox will be updated. Therefore upper limit of the size is controlled by ρ and given by (8).

$$R_j\le M(1-\rho^a) \tag{8}$$

Hyperbox represented by weights $w=(w_1,w_2,1-w_3,1-w_4)$ has corners, input patterns a_1 inside and a_2 outside the box is shown in **Fig. 2**. **Fig. 3** illustrates and provides the truth that a pattern that lies in two hyperboxes will be won by a box whose size is small and have large weights. In addition to ensuring better match, match tracking mechanism also causes category proliferation due to class overlap, pattern presentation order and presence of noise in data.

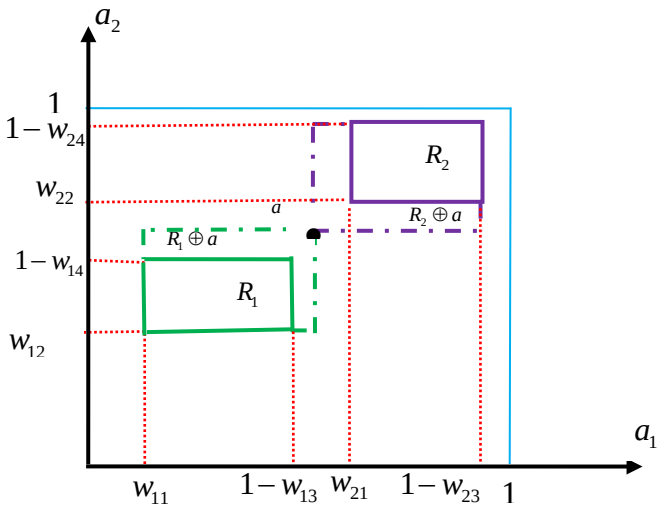


Fig. 4. Geometrical representation of two hyperboxes (R_1 and R_2) associated for Fuzzy ART categories in a two dimensional input space. For an input a either R_1 or R_2 has to be enlarged to accommodate a, such that $R_1 \oplus a > R_2 \oplus a$ for $a \to R_1$, $R_2 \oplus a > R_1 \oplus a$ for $a \to R_2$

3.2.3.1 Problem of overlapping classes

FAM network tries to classify all input patterns during training. But a large number of small categories are formed at regions, where two classes overlap. These categories will not contribute to predictive accuracy, since patterns in overlapping region may belong to either class. This leads to creation of many hyperboxes. During learning, class mismatches may occur and match tracking increases vigilance parameter. Higher value of vigilance parameter causes difficulties for existing categories to pass the vigilance test. Hence a new category is created to represent a particular pattern. Presence of large number of categories causes category proliferation problem.

3.2.3.2 Problem due to order of presenting

ART^a module performs an unsupervised clustering of input patterns. Correctness of category that resonates to an input pattern is examined by match tracking mechanism. This ensures that if the same pattern is presented subsequently, same category has to be selected. If selected category J does not provide a better match, ρ^a is increased, J^{th} category is reset and a new category is selected say k. It is verified that $|R_J \oplus a| > |R_k \oplus a|$. After learning, $|R_k \oplus a|$ will be the smallest hyperbox containing the input pattern. If the pattern is selected during subsequent instances, the system will select this category k.

Consider each category in **Fig. 4** represented by regions R_1 and R_2 has a different associated class label. Let a_1 belongs to class predicted by R_1. If pattern a_1 is presented R_2 is selected as it offers higher choice value. Since R_2 predicts a wrong class, match tracking mechanism is triggered by incrementing ρ^a to a value such that $\rho^a > \frac{|I^a \wedge w_2|}{I^a} = 0.7$. R_2 is reset and then category R_1 is evaluated. As ρ^a has been incremented R_1 also will not meet the

vigilance criteria $\frac{\left|I^a \wedge w_1\right|}{I^a} = 0.6 < 0.7 < \rho^a$ and thus R_1 is also reset. A new category is created and a_1 is assigned to it. If R_2 is not created and baseline vigilance $\overline{\rho^a} = 0$ i.e. if input patterns that have created R_2 has not presented to the system before a_1, then a_1 will select the class predicted by R_1. Thus match tracking mechanism that ensures prediction accuracy also can cause category proliferation based on order of presentation of input patterns.

4. Non-Proliferation Fuzzy ARTMAP

NPFAM architecture is a modified version of FAM. Proposed NPFAM includes an inter ART reset mechanism which avoids category proliferation problem and also does not raise ART^a vigilance. But by performing an off-line learning stage, predictive accuracy is ensured. One-to-many $F_2^a \to F_2^b$ relationships are allowed and their probabilistic information is stored in w_{jk}^{ab}. An offline map field with weight v_{jk}^{ab} is included in FAM. v_{jk}^{ab} stores probability of $F_2^a \to F_2^b$, when inter-ART is reset during prediction mode. Entropy of training set is projected by the system using these weights. Also a vigilance parameter is assigned to each category node in ART^a.

4.1. Learning Phase

Before training, all weights are initialized to one. But w_{jk}^{ab} is initialized to zero; $j = 1, 2, ..., N^a, k = 1, 2, ..., N^b$. A baseline parameter ($\overline{\rho^a}$) is set as starting vigilance. $\overline{\rho^a}$ is set to zero to minimize the number of categories. Vigilance parameter is raised in case the input space demands. In addition two parameters $h_{\max}$ and $H_{\max}$ is defined to fix upper limit of h_j and H respectively for restricting growth of hyperboxes. After initialization and identifying active node, training proceeds with vigilance test, entropy test and finally learning is completed with offline evaluation.

4.1.1. Vigilance Test

Training is continued by presenting input-output pairs, (a,b) to the network. When a pattern {a} is presented to ART^a, category J is selected according to (1). If the category is newly committed i.e. the category is not assigned to any other input pattern previously then $\rho_J = \rho_a$. Vigilance test is performed to evaluate the reset condition using ρ_J in (4). If category J fails vigilance test, this node will be suspended and new search is initiated. Pattern {b} is presented to ART^b and category k, that best matches to pattern {b} is selected. Then match field activity vector x^{ab} is formed by (7).

4.1.2 Entropy Test

Smaller hyperboxes representing the categories will provide high activation values. The part of the hyperbox, which is not available or contribute in predictive accuracy, is called entropy. The size of the entropy has to be restricted to ensure efficient classification and also avoids overlapping. The total entropy H is given by,

$$H = -\sum_{j=1}^{N^a} P_j \sum_{k=1}^{N^b} P_{jk} \log P_{jk} \tag{9}$$

Probability of occurrence of A_j is denoted by P_j and P_{jk} is the probability of B_k assuming A_j. Contribution of j^{th} node to total entropy h_j is given by,

$$h_j = -P_j \sum_{k=1}^{N^b} P_{jk} \log P_{jk} \tag{10}$$

and $$H = -\sum_{j=1}^{N^a} h_j \tag{11}$$

From activation vector x^{ab} of map field, h_j is calculated. If J^{th} node is selected then

$$P_{jk} = \frac{x_k^{ab}}{\left|x^{ab}\right|} \quad \text{if } j = J$$

$$P_j = \frac{\left|x^{ab}\right|}{\left|x^{ab}\right| + \sum_{i=1, i\neq j}^{N^a} \left|w_i^{ab}\right|}$$

$$P_{jk} = \frac{w_{jk}^{ab}}{\left|w_j^{ab}\right|} \text{ otherwise}$$

$$P_j = \frac{w_j^{ab}}{\left|x^{ab}\right| + \sum_{i=1, i\neq j}^{N^a} \left|w_i^{ab}\right|} \tag{12}$$

If $h_j > h_{\max}$ then J^{th} node in ART^a is inhibited by setting $T_J(I^a) = 0$, without raising vigilance parameter. Search process is continued to choose the other categories in ART^a, till entropy condition $h_j < h_{\max}$ is satisfied.

4.1.3 Winner test

An input pattern is classified belonging to a particular hyperbox (winner node), if the pattern lies inside the box (provides higher activation value) or the box enlarges to include the pattern. Any algorithm must have the capability to restrict the growth of hyperboxes to include a single pattern that is away from existing patterns committed to the hyperbox. This ensures predictive accuracy by restricting the size of the box. This is achieved by verifying whether the node that has won the pattern satisfies (13).

$$w_J^a - \frac{\left|I^a \wedge w_j^a\right|}{M^a} \leq \delta \tag{13}$$

Value of δ is predefined and called as decision parameter. If input pattern fails the test, winner node cannot enclose the pattern. Other nodes are restricted from learning the pattern. This pattern is categorized as unlearned. Other input patterns are presented for learning. After all patterns have been presented, many hyperboxes must have been created to cluster input patterns. Any one of the newly created hyperbox may provide high activation value to patterns in unlearned category. Now patterns in unlearned category are presented in subsequent iterations. If still some patterns are left out unlearned after a specified number of iterations,

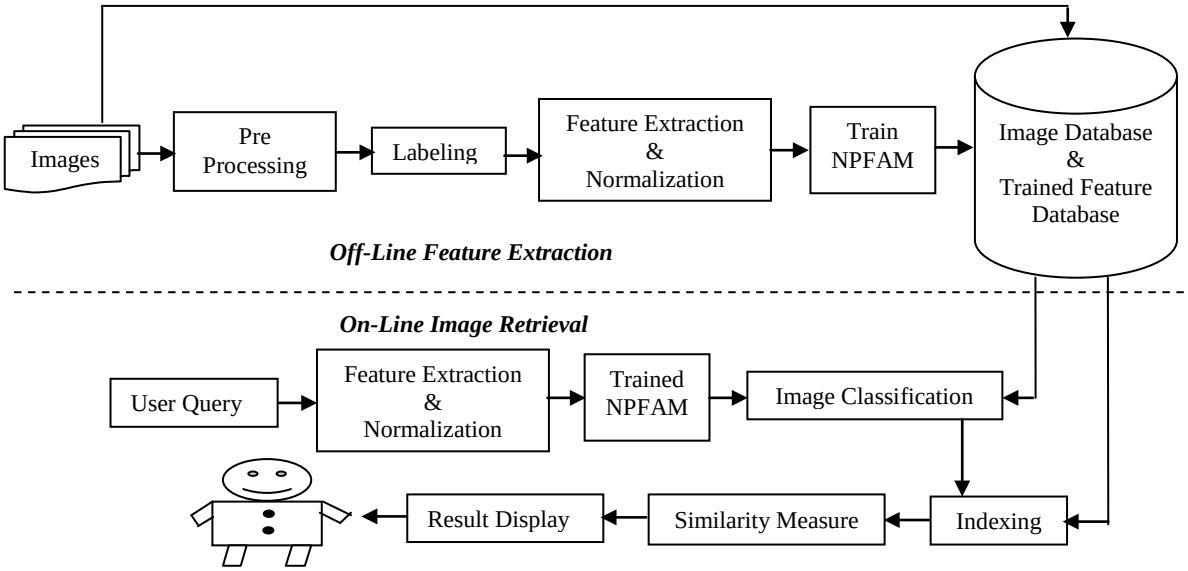


Fig. 5. Conceptual framework for Content Based Image Retrieval using NPFAM Classifier

those are assigned to a new category. This largely reduces classification error and creation of new categories. Parameter δ prevents overlapping and avoids dependency of NPFAM on training pattern presented.

4.1.4 Offline Evaluation

Once training sets are learned by the nodes in output layer, learning process is considered to be completed. After processing all patterns, offline mapping field weights $v_{jk}^{ab} = 0 : j = 1, 2, ..., N^a; k = 1, 2, ..., N^b$ and training data is presented to revise these weights. The units from ART^a and ART^b is selected in an unsupervised manner and weights v_{jk}^{ab} is updated. Total entropy H is computed by (9) with value of P_{jk} and P_j computed from (14).

$$P_{jk} = \frac{v_{jk}^{ab}}{\left|v_j^{ab}\right|}; P_j = \frac{v_j^{ab}}{\sum_{i=1}^{N^a} \left|v_i^{ab}\right|} \tag{14}$$

Impurity of all category nodes constructed during training process is described by $H_{\max}$. It also controls overtraining and influences overall accuracy of NPFAM classifier. If $H > H_{\max}$, then mapping defined by NPFAM between input and output portions are too large. Input spaces are partitioned further to bring them within $H_{\max}$ and to improve accuracy of the system.

To achieve the requirement, node j in ART^a with maximum value of h is identified and removed. Weights w_j^a and w_j^{ab} are set as 1 and 0 respectively. Baseline vigilance parameter ρ^a is re estimated as given by (15).

$$\rho^a = \frac{\left|w_j\right|}{M_a} = \min(1, \frac{\left|w_j\right|}{M_a}) + \Delta\rho \tag{15}$$

Newly created categories will thus be smaller in size, which is bounded by (9).

Training:	**Testing:**
1. *Initialize all NPFAM parameters* 2. *Select training set patterns* $a_i = (a_{i1}, a_{i2})$ *and find complement coded input to* ART^a *module* $I_i^a = \{a_{i1}, a_{i2}, 1-a_{i1}, 1-a_{i2}\}$ 3. *Calculate activation value* $T_j(I_i^a)$ *and identify the active node* J *in* ART^a *and* J' *in* ART^b 4. *Perform vigilance test on the active node if* J *passes the test.* 5. *If* $J = J'$ *then perform entropy test and winner test. If* I_i^a *passes the two tests assign* I_i^a *to* J *.* 6. *If* $J \neq J'$ *reset* $T_j(I_i^a) = 0$ *and initiate match tracking.* 7. *After all the patterns in training set are learned, offline evaluation is done.* 8. *Repeat steps 2 to 7 up to specified number of iterations or if the error reaches a defined threshold*	1. *The query image is issued for searching.* 2. *Activation value is estimated for all the categories.* 3. *The pattern is labeled with class corresponding to category that yields highest activation value.* 4. *The Euclidean distance between the query image feature and between images of particular class in database is estimated.* 5. *The images are displayed to user in ascending order of distance measure.*

Fig. 6. Algorithm for Training and Testing Phases

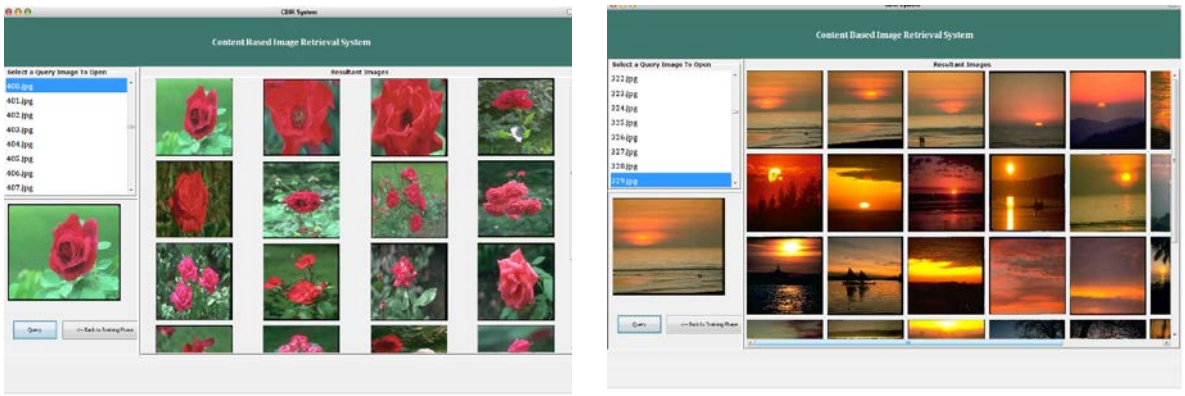
is carried out till $H < H_{\max}$. Patterns of deleted categories are submitted in next iterations. If $H_{\max} = 0$ then training algorithm takes long time to complete. If $H_{\max}$ is set to high value, the process will be terminated earlier and leads to over simplification. $\Delta\rho$ used in (15) ensures, the category with $H > H_{\max}$ is not created after getting detected. Thus input space is partitioned into categories such that retrieval accuracy is preserved.

4.2 Retrieval System

The proposed algorithm is implemented in two phases. Training phase in offline and testing phase in online. The conceptual framework with proposed NPFAM algorithm is depicted in **Fig. 5**. The algorithm depicting both the process is given in **Fig. 6**.

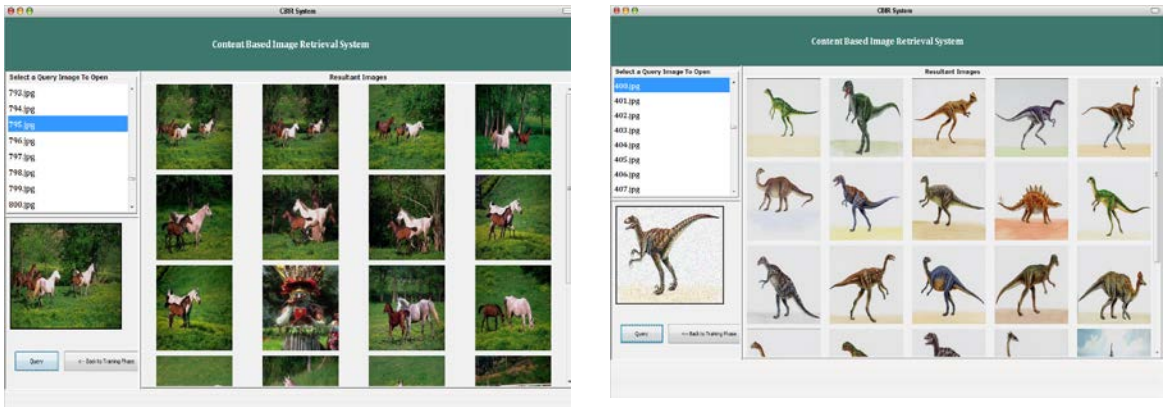
4.2.1 Training Phase

This phase constitutes extraction of features from images and scaling them between 0 and 1. Scaled features are stored in the database as feature vectors. The proposed system, with NPFAM classifier is trained with randomly sampled images across all classes and is labeled. The feature vector of each labeled test set image forms training vector. Thus NPFAM is trained to labeled classes by fusing training vector into the classifier.



(a) (b)

Fig. 7. Retrieval results for Corel 10k query images. a) Rose b) Sunset



(c) (d)

Fig. 8. Retrieval results for Corel 1k query images. c) Horse, d) Dinosaurs with Gaussian Noise(Mean=0.1,Variance=0.04)

4.2.2 Testing Phase

Feature vectors of all images in database are presented to the trained NPFAM classifier. The classifier does the mapping between input vectors and classes. During testing phase, query image is provided as input. Feature vector is extracted and normalized between 0 and 1. The classifier identifies the class of the query image. Images in the identified class are evaluated using (16) and displayed in increasing order based on similarity.

$$Dist(I_d, I_q) = \sum_{i=1}^{N_{fd}} Dist(I_d^{fi}, I_q^{fi}) w_{fi} \tag{16}$$

where N_{fd} is number of feature descriptors and

$$Dist(I_d^{fi}, I_q^{fi}) = \frac{1}{p} \sum_{k=1}^{p} x_p - y_p \tag{17}$$

where p = number of elements in feature vectors corresponding to each descriptor.

Table 1a. Precision (P) and Recall (R) Values for Corel 1k data set

Query Image Class	Total retrieved images	Relative retrieved images	P	R
Africans	96	85	0.885	0.85
Beach	93	82	0.881	0.82
Building	92	80	0.869	0.80
Bus	95	81	0.852	0.81
Dinosaurs	90	88	0.977	0.88
Rose	91	88	0.967	0.88
Elephants	89	83	0.932	0.83
Horses	92	87	0.945	0.87
Mountains	90	82	0.911	0.82
Food	90	84	0.933	0.84

Table 1b. Precision (P) and Recall (R) Values for selective classes among100 classes in Corel 10k data set

Query Image Class	Total retrieved images	Relative retrieved images	P	R
Butterfly	94	81	0.862	0.81
Beach	96	79	0.823	0.79
Sunset	95	80	0.842	0.80
Cars	94	74	0.787	0.74
Waterfalls	97	78	0.804	0.78
Rose	95	81	0.853	0.81
Elephants	96	76	0.792	0.76
Horses	96	82	0.854	0.82
Mountains	94	75	0.798	0.75
Food	95	80	0.842	0.80

5. Experimental Results

The proposed system is developed using Java and My SQL and implemented to retrieve general purpose image database from COREL. Two Image databases are used to validate and test the System. Data set1 having 10 classes with 100 images in each class (total 1k images) and Data set2 having 100 classes with 100 images in each class (total 10k images) in JPEG format of size 384X256 or 256X384. Raw images are used to validated and test. Visually similar images are considered to be an entity in a particular class. A maximum of 10 images from each class selected randomly is considered for training. Retrieval performance of the system from both data sets is evaluated using standard measures Precision (P), Recall (R) and F-Score. F-score is the harmonic average of recall and precision which reflects the degree of similarity and order of answers. Retrieval accuracy for query images with Gaussian noise of 0.1 mean, 0.04 variance is also verified.

5.1 Query Examples and System Demonstration

For evaluating the system performance a query image is submitted to the system. NPFAM classifier identifies the class of the query image. Feature vector of images in the particular class and query image are matched using similarity metric. The most similar images are displayed to the user. In every experiment, the system is evaluated by five images selected at random from each class as query image. **Fig. 7** and **Fig. 8** show the display of the query image and retrieved images for the class Rose and Sunset from Corel 10k data set, Horse and Dinosaur (with Gaussian noise in the query image) from Corel 1k data set.

5.2 Retrieval Precision/Recall and F-Score

To evaluate the effectiveness of the proposed approach the retrieval performance can be defined using Precision (P) and Recall (R). Average value of results obtained in five runs

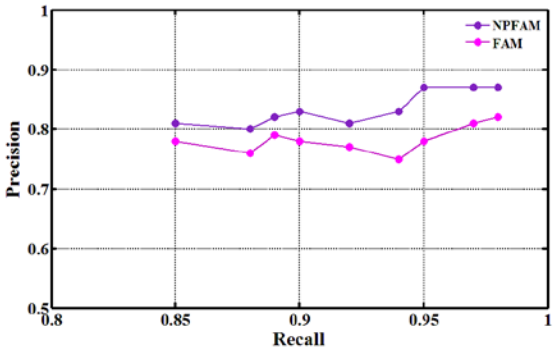


Fig. 9. Precision Recall Plot for FAM and NPFAM

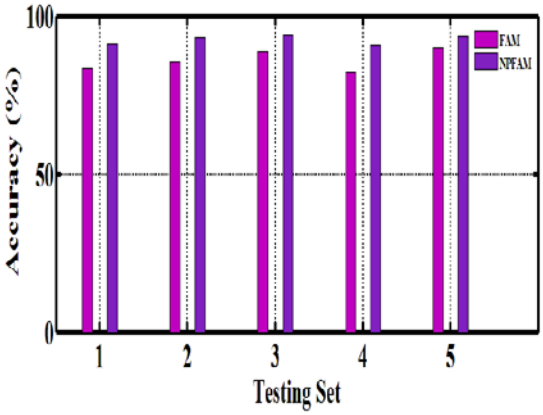


Fig. 10. Classification Accuracy for FAM and NPFAM

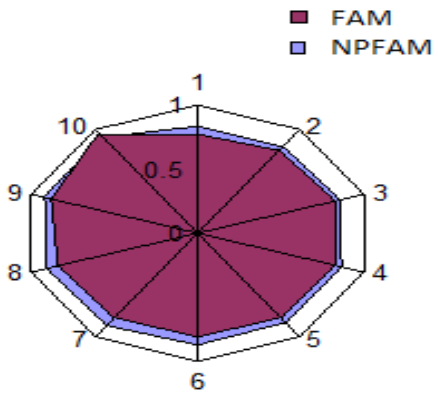


Fig. 11. F-Score Plot for FAM and NPFAM

of different query images selected at random from a particular class is reported. For a particular query image, relevant images are considered to be those that belong to the same class as that of the query image. P and R is calculated by (18).

$$P = \frac{NI_A(q)}{NI_R(q)}; R = \frac{NI_A(q)}{NI_t} \tag{18}$$

where $NI_A(q)$ is the number of retrieved images relevant to the query image. $NI_R(q)$ is number of images actually retrieved and NI_t is number of images in the database which are relevant to the query image. **Table 1a** and **1b** provides the precision and recall values for all 10 classes in data set 1 and selective 10 classes in data set 2.

Fig. 9 gives the precision at different recall values for FAM and NPFAM**. Fig. 10** shows the comparison between the classification accuracy for different sets of images selected at random using FAM classifier and NPFAM classifier. From the results it is evident that NPFAM performs better than its unmodified version. **Fig. 11** provides the F-Score plot for both methods. F-Score is estimated by (19).

$$F = \frac{2 \times P \times R}{P + R} \tag{19}$$

Table 2. FAM and NPFAM Parameters used in the experiments

Training Set	Architecture	H_{max}	h_{max}	Δρ	N^a	T(s)
Set 1	FAM				13.3	120
	NPFAM	0.0	1	0.02	6.0	100
Set 2	FAM				6.5	132
	NPFAM	0.0	1	0.05	3.0	113
Set 3	FAM				35	121
	NPFAM	0.1	1	0.05	11	105
Set 4	FAM				25	119
	NPFAM	0.1	1	0.02	8.2	102
Set 5	FAM				14.6	140
	NPFAM	0.2	1	0.02	7.4	123

Table 3. Comparison of Accuracy between proposed NPFAM and other methods

Method	Retrieval Accuracy(%)
Proposed NPFAM	94.6
Multiple Feature Approach [7]	96.6
FRAR [8]	84.97
Adaptive FCH [9]	82.5
CMR scheme [10]	92.65
IGA [11]	80.6
FAM [23]	90.4

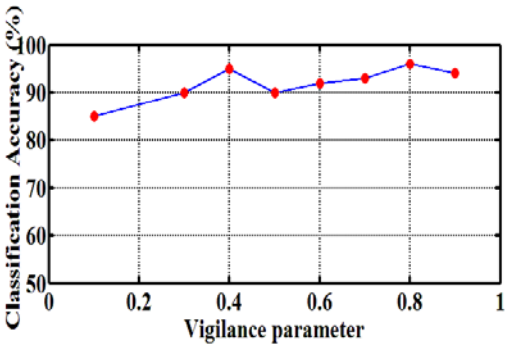


Fig. 12. Accuracy of NPFAM classifier for different values of Vigilance Parameter

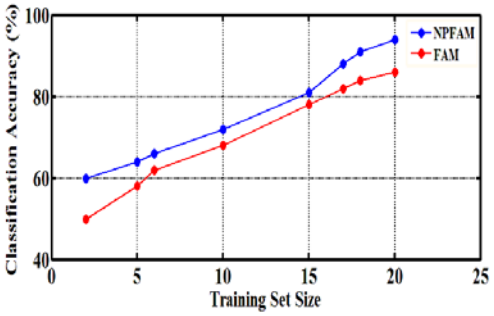


Fig. 13. Classification Accuracy Vs Training set size

5.3. NPFAM Parameters

The retrieval results of the system depend on NPFAM parameter values. **Table 2** gives the optimal parameter values used in experiments for various generated categories (differing in number) in all training sets. Each set consists of 20 images selected at random from the database. A FAM-based system highly depends on the order of training pattern. Thus, large number of categories would be required. This produces more overlapping categories leading to misclassification and larger network. Developing a optimum network, to ensure better performance with a smaller network is a seperate task.

Moreover, redundant categories are generated as it raises ρ^a after implementing Inter-ART process. In NPFAM, ρ^a is raised in offline mode and can be ignored when not necessary. This guides classifier to find optimal number of categories. Also, NPFAM is observed to be less susceptible to order of input pattern presented during training. **Fig. 12** provides accuracy of the proposed system for various values of vigilance parameter. Optimum results are shown to be achieved when vigilance parameter is assigned to 0.2, 0.4 and 0.8. Comparison of NPFAM with other methods is provided in **Table 3.** Retrieval accuracy for proposed technique is achieved using raw images for querying, classification and matching.

Table 4a. CPU Time of NPFAM and FAM for Corel 1k data set

Query Image Classes	CPU Time(in milliseconds)	
	NPFAM (Global Image Search)	FAM (Region-Based Search)
Africans	1800	1900
Beach	1600	1950
Building	1500	2000
Bus	1240	1800
Dinosaurs	1000	1950
Rose	900	1770
Elephants	1400	2100
Horses	1300	1600
Mountains	1270	1820
Food	1460	2050

Table 4b. CPU Time of NPFAM for selective classes among100 classes in Corel 10k data set

Query Image classes	CPU Time(in milliseconds)
	NPFAM (Global Image Search)
Butterfly	1920
Beach	1850
Sunset	1200
Cars	1450
Waterfalls	1820
Rose	1240
Elephants	1800
Horses	1570
Mountains	1640
Food	1790

5.4 Speed

NPFAM has been implemented on pentium III 700 MHz PC running on Linux operating System. Speed of proposed CBIR system using NPFAM is compared with that using FAM [23]. Proposed system reduces search space size by labeling images prior to retrieval. This speeds up querying process and enables easy and effective searching of large image database. **Fig. 13** gives accuracy of NPFAM and FAM classifiers used in a CBIR system for different sizes of training sets. It has been observed that system accuracy is directly proportional to training set size. This is due to expansion and sharpness of training boundaries, resulting from the large number of training sets. **Table 4a** gives a comparison of the CPU time for NPFAM and FAM based systems. **Table 4b** provides the CPU time for Corel 10k data set in NPFAM system.

6. Conclusion and Future Enhancement

The implementation of proposed system includes a comfortable GUI. The CBIR system with proposed NPFAM classifier is used to retrieve similar images from database. The classifier generates small categories assuring high accuracy. The work has also experimented influence of values assigned to parameters on system performance. Results achieved prove that NPFAM classifier is a solution to category proliferation problem, yields good results for query images with noise and also without implementing region-based classification. Proposed algorithm also produces good results for different data sets and is independent on training pattern presentation. NPFAM based retrieval system uses a large size of feature vector. It is successfully fused into classifier, trained and minimum target error is achieved. The system performance can be improved for small training sets by including relevance feedback.

References

[1] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain,"Content-based image retrieval at the end of the early years.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1349–1380, 2000. Article (CrossRef Link)

[2] Rui, Y., Huang, T.S., Chang, S. F., "Image Retrieval: Current Techniques, Promising Directions and Open Issues,"*Journal of Visual Communication and Image Representation, Transaction on Systems, Man, and Cybernetics*, vol. 8, pp. 460–472, 1999. Article (CrossRef Link)

[3] Thomas Sikora, "The MPEG-7 Visual Standard for Content Description—An Overview," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, 2001. Article (CrossRef Link)

[4] Berchtold,C. Bohm,D. Keim, and H. Kriegel, "A cost Model for nearest neighbor search in high-dimensional data space," in *Proc. of ACM Symp. on Principles of Database Systems*, pp 78-86, Tuscon, Arizona,1997. Article (CrossRef Link)

[5] K.V. Ravi Kant, Divyakant Agrawal, Amr El Abbadi, Ambuj Singh, "Dimensionality Reduction for Similarity Searching In Dynamic Databases," *Computer Vision and Image Understanding: CVIU*, 1998. Article (CrossRef Link)

[6] M. Uysal, F. Y. Vural, "Selection of the Best Representative Feature and Membership Assignment for Content-Based Fuzzy Image Database," in *Proc. of CIVR'03 Proceedings of the 2nd international conference on Image and video retrieval* pp. 141-151, 2003. Article (CrossRef Link)

[7] Yasmin Mussarat, Sharif Muhammed, Moshsin Sajjad and Irun Isma, "Content Based Image Retrieval using combined features of shape, color and relevance feedback," *KSII Transactions on Internet and Information Systems*, vol. 7. no. 12, 2013. Article (CrossRef Link)

[8] K. Seetharaman, S. Sathiamoorthy, "Color image retrieval using statistical model and radial basis function neural network," *Egyptian Informatics Journal*, vol. 15, 2014. Article (CrossRef Link)

[9] Esmat Rashedi, Hossein Nezamabadi-pour, Saeid Saryazdi, "A simultaneous feature adaptation and feature selection method for content-based image retrieval systems," *Knowledge-Based Systems*, vol. 39, 2013. Article (CrossRef Link)

[10] Sudipta Mukhopadhyay, Jatindra Kumar Dash, Rahul Das Gupta, "Content-based texture image retrieval using fuzzy class membership," *Pattern Recognition Letters*, vol. 34, 2013. Article (CrossRef Link)

[11] Chih-Chin Lai, Ying-Chuan Chen, "A User-Oriented Image Retrieval System Based on Interactive Genetic Algorithm," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, 2011. Article (CrossRef Link)

[12] S. Grossberg, "Adaptive pattern classification and universal recoding ii: feedback, expectation, olfaction, and illusions," *Biological Cybernetics* pp. 187–202, 1976. Article (CrossRef Link)

[13] Gail A. Carpenter, Stephen Grossberg, Natalya Markuzon, John H. Reynolds and David B. Rosen, "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, 1992. Article (CrossRef Link)

[14] Vassilis .G . Kaburlasos, Stelios . E . Papadakis, and George. A .Papakostas, "lattice computing extension of the fam neural classifier for human facial expression recognition," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 10, 2013. Article (CrossRef Link)

[15] Andrey makrushin, Claus vielhauer, Jana dittmann, "The Impact of Artmap to Appearance-based Face Verification," in *Proc. of the 12th ACM workshop on Multimedia and security*, pp. 89-94, 2010. Article (CrossRef Link)

[16] A.Kaylani, M.Georgiopoulos, M.Mollaghasemi, G.C.Anagnostopoulos, "AG-ART:An adaptive approach to evolving ART architectures," *Neurocomputing*, vol. 72, 2009. Article (CrossRef Link)

[17] Gail A. Carpenter, Boriana L. Milenova, Benjamin W. Noeske, "Distributed ARTMAP: a neural network for fast distributed supervised learning," *Neural Networks*, 1998. Article (CrossRef Link)

[18] Shing Chiang Tan a, M.V.C. Rao, Chee Peng Lim, "Fuzzy ARTMAP dynamic decay adjustment: An improved fuzzy ARTMAP model with a conflict resolving facility," *Applied Soft Computing*, 2008. Article (CrossRef Link)
[19] Stephen J. Verzit , Gregory L. Heilemanj, Michael Georgiopoulus, Michael J. Healy, "Rademacher Penalization Applied to Fuzzy ARTMAP and Boosted ARTMAP," *IEEE*,2001. Article (CrossRef Link)
[20] E. Gomez-Sanchez, Y.A. Dimitriadis, J.M. Cano-lzquierdo, J. Lopez-Coronado, "Safe µARTMAP: A new solution for reducing category proliferation in Fuzzy ARTMAP," *IEEE*, 2001. Article (CrossRef Link)
[21] Wing Yee Sit, Lee Onn Mak, Gee Wah Ng, "Managing Category Proliferation in Fuzzy ARTMAP Caused by Overlapping Classes," *IEEE Transactions on Neural Networks*, vol. 20, no. 8, 2009. Article (CrossRef Link)
[22] B. S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, Akio Yamada, "Color and Texture Descriptors," *IEEE Transactions On Circuits And Systems For Video Technology*, vol. 11, 2001. Article (CrossRef Link)
[23] Mutlu Uysal, Yarman Vural, "A Content-Based Fuzzy Image Database Based on the Fuzzy ARTMAP Architecture," *Turk J Elec Engin*, vol.13, no.3, 2005. Article (CrossRef Link)

Anitha Kannan
Assistant Professor, R.M.K Engineering College, Chennai
Ph.D. Student, Anna University, Chennai
M.E. Anna University, Chennai
Research Interests:
Image Retrieval, Pattern Recognition and Machine Learning

Chilambuchelvan Arul Gnanaprakasam
Professor, R.M.K Engineering College, Chennai
Ph.D. College of Engineering, Anna University, Chennai
M.E. Coimbatore Institute of Technology, Coimbatore
Research Interests:
Embedded system, VLSI design and soft computing