An Effective Mechanism of Feature Based Retrieval of Mathematical Expression from Documents

S. Jhansi Rani
Assistant Professor, Department of Computer Science & Systems Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India.

Dr. V. Valli Kumari
Professor, Department of Computer Science & Systems Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India.

Abstract
Nowadays, Content Based Image Retrieval (CBIR) plays a significant role in the image processing field. Images relevant to a given query image are retrieved by the CBIR system utilizing either low level features (such as shape, color etc.,) or high level features (human perception). Image-based math retrieval is a new area of research which has gained importance because of the need for extracting the mathematical expressions for processing. In this paper, a method for locating mathematical expressions in document images without the use of optical character recognition is presented. Initially, when a query image is given, images relevant to it are retrieved from the image database based on its feature values. We have performed retrieval utilizing one of the evolutionary algorithms called Evolutionary Programming (EP). Subsequent to this process, query keyword which is a generally feature values is extracted from these retrieved images and then based on this query keyword, relevant images are retrieved from the database. The images retrieved based on feature values are compared and the images which are both visually and semantically identical are identified. Better results obtained by the proposed approach when it was compared to existing method for Heterogeneous Document Images, it is queried using different types of images prove the efficiency of the implemented technique.

Introduction
Visual information has been used in a wide-range of fields as a result of the rapid development of digital imaging and networking technologies. The necessity to efficiently retrieve such information from databases has increased the interest in image content based image retrieval techniques. Most of the approaches which have been proposed in the past for image retrieval use Content-Based Image Retrieval (CBIR) methods which retrieve images by means of keywords or image contents [1]. In image processing, CBIR is considered as an important domain because of its diverse applications in internet, multimedia, medical image archives, satellite imaging [16], commerce, government [9], and academia and crime prevention [6]. Images are indexed by CBIR using the image features like color, shape and texture with minimal human involvement[8]. CBIR compares the features of the images present in the database with that of the query image for retrieving relevant images [4] from the image data base for a specified query image [5]. CBIR has been developed into a popular field of research. Determining the finest solution for efficient retrieval of multimedia data and utilization of storage space has been attempted by several research domains [2]. Images are retrieved using the relevance that is computed based on the similarity of the features between them and the query image [3] [11].

In recent years, the PDF format has become widely accepted as a quasi-standard for document presentation and exchange. Document analysis starting directly from PDF documents is therefore becoming increasingly important. Research in this area currently ranges over extracting and identifying various components of the documents. The output can then be used for purposes such as improving search and indexing capabilities, or converting into other formats such as XML. Some of this work exploits information contained in the PDF file, whilst others only work with a rasterised version of the file [10]. As large amounts of technical documents have been published in recent years, efficiently retrieving relevant documents and identifying locations of targeted terms are urgently needed. We have already widely utilized search engines like Google to find technical documents. However, currently only text-based keywords are used for retrieving documents having related text in title, abstract, or main body.

We argue that mathematical formulas have been overlooked in technical document retrieval for a long time. Scientists severely express their ideas in math, following some conventions or unwritten customs to define notations, which make retrieving technical documents by math a feasible idea [12].

There are several ways how to create and publish semantically annotated mathematical content. However, these documents are still a minority of the mathematical content on the WWW. Among the commonly used document formats to exchange mathematics only MathML contains support for semantics [14]. Mathematical formula identification is a critical step in mathematical notation recognition and scientific document management. It aims at detecting and segmenting mathematical formula regions from the document pages. In the past decade, a number of mathematical formula identification algorithms have been reported. However, there are few direct performance comparisons of the different methods due to the following obstacles in ground-truth dataset, evaluation metric, and automatic evaluation methods.
[15]. Mathematical formulas are classified into two categories: 1) isolated formulas printed in separate lines; 2) embedded formulas mixed with ordinary text. Existing isolated formulas identification approaches are rule-based and machine learning-based. Since the isolated formulas exhibit distinct geometric layout features, it is relative easier to identify them. It is observed that some isolated formula identification techniques provided rather high accuracy. In contrast, the embedded formula identification is more challenging, because the embedded formulas are generally short expressions, which are difficult to discriminate from ordinary text. Consequently, the lower accuracy is expected for embedded formula identification compared with isolated formula identification [17].

Recent Related Researches: A Review
A handful of research works available in the literature are briefly reviewed in this section. Hristidis et al. [18] have presented algorithms that return the top results for a query, ranked according to an IR-style ranking function, while operating on top of a source with a Boolean query interface with no ranking capabilities. The algorithms generated a series of conjunctive queries that return only documents that are candidates for being highly ranked according to relevance metric. Their approach could also be applied to other settings where the ranking was monotonic on a set of factors (query keywords in IR) and the source query interface was a Boolean expression of these factors. Their comprehensive experimental evaluation on the PubMed database and a TREC data set show that they achieved order of magnitude improvement compared to the current baseline approaches.

Garain et al. [19] have proposed an automatic understanding of online handwritten mathematical expressions (MEs) written on an electronic tablet. The proposed technique involves two major stages: symbol recognition and structural analysis. Combination of two different classifiers has been used to achieve high accuracy for the recognition of symbols. Several online and offline features are used in the structural analysis phase to identify the spatial relationships among symbols. A context-free grammar has been designed to convert the input expressions into their corresponding TEX strings which are subsequently converted into MathML format. Contextual information has been used to correct several structure interpretation errors. A method for evaluating performance of the proposed system has been formulated. Experiments on a dataset of considerable size strongly support the feasibility of the proposed system.

Awais Adnan et al. [13] have proposed an object based search technique where geometrical shapes and other features like color and texture have been used to identify the object in the image. Also, the search process has been enhanced by object-co-relation augments. Though, they have chosen simple images to decrease the role of segmentation in the overall process in order to focus more on object identification, the same method has been applicable to other images also.

Jalil Abbas et al. [7] have proposed content based image retrieval and mainly concentrated on Text Based image retrieval (TBIR). Comparison of their results have shown that content based has been visual whereas text based has been semantic. Compared to Content Based image retrieval (CBIR), the text based image retrieval has been faster.

Proposed Approach For Mathematical Formula Retrieval Using Evolutionary Programming
The content based image retrieval has become an interesting research topic in recent years due to the vast availability of data’s and documents as a source of information. Various data like images, text, mathematical expression etc can be retrieved from the internet source using certain retrieval techniques. In our proposed method we have developed an efficient retrieval system of mathematical expression from the document with the aid of soft computing techniques. The document from which the expression has to be retrieved is given as the input for the system. For accurate retrieval process to be carried out, we have filtered the document using median filter inorder to remove the noise that is present. The mathematical expression is then separated from the document using background elimination techniques. Once the expression is separated from its background, feature extraction is carried out which involves extraction of features like pixel count and local gradient histogram features. Once these features are extracted, these feature vectors are given as input to the evolutionary programming algorithm. In EP, the optimizations of these features are carried out and the expressions are retrieved based on the fitness value. The general flow diagram of our proposed retrieval system is shown in the fig 1 below.

Noise removal using Median filter:
The initial process we employed in our proposed method is the noise removal process. In our proposed method we utilize

![Figure 1: Basic Flow diagram of proposed retrieval method.](image-url)
median filter for the noise removal. The median filter is often applied to gray value images due to its property of edge preserving smoothing. In the median filtering operation, the pixel values in the neighborhood window are ranked according to intensity, and the median becomes the output value for the pixel under evaluation. The process is follows:

In median filtering, the neighboring pixels are ranked according to brightness and the median value becomes the new value for the central pixel. Median filters can do an excellent job of rejecting certain types of noise, in particular, “shot” or impulse noise in which some individual pixels have extreme values. The general expression for the median filter is given as per the below eqn,

\[ M_f(a_1, a_2, ..., a_N) = \text{MIN}\left(\sum_{i=1}^{N}|a_i - a|, ..., \sum_{i=1}^{N}|a_N - a|\right) \]  

Using Eqn 1, the median filtering is performed to remove the noise from the acquired image. Once the noise removal is done, the next step in our proposed method is the background elimination which is performed to separate the mathematical expressions from the document for processing. The detailed process is explained in the below section.

**Background elimination:**

Background elimination usually termed as binarization, is an essential part of preprocessing step in image processing, converting gray-scale image to binary image, which is then used for further processing such as image analysis and character recognition. Images are virtually split into small blocks either in foreground or background component. Based on intensity variation these classifications as foreground or background is done. Intensity variance of a text block is considerably more than that of a background block. If the intensity variance of a block is less than an adaptive threshold it is considered as part of the background. Otherwise, it is considered as part of a foreground component. This classification is done on the basis of intensity variance within the block. The intensity variance is defined as the difference between the maximum \( I_{\text{max}} \) and the minimum \( I_{\text{min}} \) gray scale intensity within the block. It is observed that the intensity variance of a text block is considerably more than that of a background block. This has been the key idea behind the present approach. So, if the intensity variance of a block is less than an adaptive threshold (\( T \)), it is considered as part of the background. Otherwise, it is considered as part of a foreground component.

The threshold \( T \) has two components, i.e., a fixed one (\( T_f \)) and a variable one (\( T_v \)) as shown in eqn. 2. \( T_f \) is a constant subject to tuning. The formulation of \( T_v \) is given in Eq. 3. It may be noted that \( I_{\text{min}} \) must be greater than a heuristically chosen threshold \( T_{\text{min}} \). Otherwise, the block is considered as part of a foreground object. This reveals the reality that even if the intensity variance of a block is less than \( T_0 \), it is not classified as background until the minimum intensity within the block exceeds \( T_{\text{min}} \). This reduces the possibility of misclassifying foreground blocks as background ones.

\[ T_0 = T_f + T_v \]
\[ T_v = \left(\left[I_{\text{min}} - T_{\text{min}}\right] - \min\left(T_f, I_{\text{min}} - T_{\text{min}}\right)\right) \cdot 2 \]  

Where,

\( I_{\text{min}} \) is the minimum intensity in a block

\( T_{\text{min}} \) is the minimum threshold value obtained

\( T_f \) is the threshold value we assign

It is evident from eqn. 3 that the computation of \( T \) is such that the more is the average intensity within the grid, the larger is the threshold. In other words, if the intensity band of the block falls towards the higher range of the overall intensity band, then \( T \) becomes larger. Such formulation helps to efficiently eliminate the background blocks from the captured business card images. Also light backgrounds get easily eliminated in the said approach. So using this technique the mathematical expressions are separated from the documents for further proceeding. Once the expressions are extracted from its background, feature extraction is carried out inorder to retrieve the expression form the documents. The next step in our proposed method is feature extraction which is explained in detail below.

**Feature Extraction:**

When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called feature extraction. Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. Various features are present in the images which can be extracted for proper differentiation of any objects from its background such as the texture, color, edges etc. In our proposed method we have utilized the pixel count feature as well as local gradient histogram features.

**Pixel count features**

- A sliding window moves from left to right over the word.
- The width of the sliding window comprises several columns.
- At each position, the height of the window is adjusted to the area actually containing pixels, and then it is split into a 4x4 cell grid.
- The pixel counts in each of these cells are concatenated to form a 16-dimensional feature vector.

To avoid boundary problems at the very first or very last positions of the sliding window, we assume the area outside the image consists of zero-valued pixels.

**Local Gradient Histogram Features:**

For each cell in the image the local gradient histogram features are calculated for extracting the features. Let \( X(a,b) \) denote result of convolving the image \( P(a,b) \) with a smoothing filter, employed for denoising purposes. For this purpose at first the horizontal and vertical gradient components \( V_x \) and \( V_y \) are determined as follows,
\[ V_a = X(a + 1, b) - X(a - 1, b) \]  
\[ V_b = X(a, b + 1) - X(a, b - 1) \]

Along with the computation of the gradient, a Gaussian derivative filter can also be obtained with a gradient magnitude \( R \) and the direction \( \phi \) for each pixel with the corresponding coordinates \((a, b)\) which is defined as follows,

\[ R(a, b) = \sqrt{V_a^2 + V_b^2} \]

\[ \phi(a, b) = \angle(V_a, V_b) \]

In the above equations, eqn 7 represents the function which returns the direction of the components \( V_a, V_b \). The magnitude of the identical orientations is accumulated into the histogram i.e., for each pixel with coordinates \((a, b)\) we determine which of the orientations are close to \( \phi(a, b) \). In these feature extractions, the pixels outside the image are assumed to be 0 in order to avoid the boundary effects.

Once the required features are extracted, the final process in our proposed method is the retrieval of mathematical expressions from different documents based on the extracted feature vectors. For this, we have employed the soft computing technique Evolutionary Programming. Here the feature vectors extracted are given as the input to the evolutionary algorithm which is an optimization algorithm used for retrieval of expressions. A detailed explanation of the retrieval process using the EP is given in the below section.

Mathematical expression Retrieval through Evolutionary Programming (EP)

When a query image is given as input to the proposed system, all the features from the query image are extracted by it. After this, the feature set of the query image is compared with each feature set present in the feature set library. For the retrieval process, in this work we utilize Evolutionary Programming (EP) which is one of the Evolutionary Algorithms. The feature set of the query image is compared with the feature set of each image present in the database using the SED (Squared Euclidean Distance) similarity measure by EP in the retrieval process. Let \( Q \) be the query image and \( \{F_q\} \) be the extracted feature set of the query image. Relevant images are retrieved by comparing the query image \( Q \) with the database images using the feature set computed for each image in the database \( D \). EP contains 3 basic steps which should repeat until it reaches its threshold or a satisfactory solution is obtained.

1. Choose an initial population of trial solutions at random. The speed of optimization is highly dependent on the number of solutions in a population, but definite answer regarding how many solutions are appropriate is not available.
2. A new population is generated by replicating each solution and each of these offspring solutions are mutated in accordance with a continuous distribution of mutation types that range from minor to extreme. The functional change forced on the parents assesses the severity of the mutation.
3. Then the fitness of each offspring solution is computed in order to assess the solution. Though traditionally the \( N \) solutions that are to be retained for the population of solutions is determined by performing a stochastic contest, sometimes it is determined deterministically. There is no necessity to keep a constant population size or to restrict the parents to have only one offspring.

Generation of random chromosomes

In our work, initially \( N_p \) numbers of random chromosomes are generated and the chromosomes represent the images and each chromosome has \( N_d \) gene values. Here the gene values indicate the indices of the images in the database and the process utilizes the images that have relevant indices. Each index may represent the image in the database. Then the generated chromosome can be represented as follows,

\[ D_l^{(j)} = \{d_0^{(j)}, d_1^{(j)}, d_2^{(j)}, \ldots, d_{N_d-1}^{(j)}\} \]

\[ 0 \leq j \leq N_p - 1 \]

Here \( D_l^{(j)} \) indicates the \( j^{th} \) chromosome and \( d \) represents the index of the image. The pixel count and the histogram features are extracted from the images whose indices are generated as genes. After computing all the above-mentioned set of features from the database \( D \), the \( s \) features of the image are concatenated as a single feature set and then each feature set is normalized as,

\[ \{\hat{F}_l^{(j)}\} = \left( \frac{\left| F_l^{(j)} \right|}{\sum_{q=0}^{\left| F_l^{(j)} \right| - 1} (F_l^{(j)}(r))^2} \right)^{1/2} \]

where,

\[ F_l^{(j)}(r) = \left| F_l^{(j)}(r) \right| - \sum_{q=0}^{\left| F_l^{(j)} \right| - 1} F_l^{(j)}(r) \]

The normalized feature set \( \{\hat{F}_l^{(j)}\} \) obtained from Eq. (9) is the final feature set extracted for a particular database image and it is stored in the feature set database. In a similar manner, the feature set for all the images are determined and a feature set library is created using these feature sets of the images.

Fitness Function:

Here the fitness utilized for this EP is SED (Squared Euclidean Distance) which is the distance measure utilized to analyze the similarity between the query image feature set and the database image feature set and the image that has the least deviated distance is considered as the image most similar to the query image. The fitness formulae computed are shown below.
\[ F^{(j)} = \frac{1}{|\delta_i^{(j)}|} \sum_{i=0}^{N_p-1} \delta_i^{(j)} \]  

(11)

where, \( \delta_i^{(j)} = \sum_{r=0}^{|T|} \left( \hat{F}_i(r) - \hat{F}_q(r) \right)^2 \)  

(12)

In eq. (12), \( \hat{F}_q \) is the feature set extracted for the query image, \( \delta_i^{(j)} \) represents the SED between each \( D_{ij}^{(j)} \) of the \( j^{th} \) chromosome and the query image, i.e., SED of the query image and the indices of database images which are generated in the genes. Subsequently, the \( F^{(j)} \) is sorted in the ascending order and \( N_p/2 \) number of mean distances are selected from \( f^{(j)} \). Then the corresponding \( D_{ij}^{(j)} \) of the selected \( F^{(j)} \) is obtained and then the selected chromosomes are subjected to the genetic operator, mutation.

**Mutation**

The mean of the chromosomes are computed and they are sorted in the ascending order and \( t \) numbers of mean values are selected for the mutation process. After that, mutation operation is performed on the chromosomes corresponding to these selected means termed as children chromosomes \( D_{next}^{(j)} \). The mutation process replaces \( N_M \) number of genes from every chromosome with the new genes. The \( N_M \) numbers of genes are nothing but genes that have minimum SED value. The replaced genes are the randomly generated genes without any repetition within the chromosome.

**Selection of Optimal Solution**

After the process is repeated \( I_{max} \) number of times, chromosomes that have maximum fitness value are selected from the resultant group of chromosomes as the best chromosomes. Here, the best chromosomes are the chromosomes that have maximum fitness. The indices, which are obtained from the genes of the best chromosomes represent the database images that are similar to the given query image and they are retrieved in an effective manner. The retrieved relevant images that have same visual content but different semantics are stored in a separate vector \( D_i \). The semantic indices of the retrieved images are analyzed and the most frequent indices are utilized as the semantic query keyword for retrieving the images. The following pseudo code details the extraction:

Input: \( N = \{ \text{id}_1, \text{id}_2, \text{id}_3, \ldots, \text{id}_{|D_L|} \} \)

Output: Query keyword

Steps:

1. **Initialize** freq is zero
2. For \( i = 0 \) to \( |N| - 1 \)
3. If (\( \text{id}_i == \text{id}_j \))
4. freq\(_i\) = freq\(_i\) + 1
5. End if
6. End for
7. End for
8. End for

The key obtained from the above pseudo code is utilized as the query keyword for the subsequent process of our proposed work, namely high level based CBIR. Thus the evolutionary programming helps in retrieving the mathematical expression form the documents which more effectiveness. The proposed technique has outperformed the existing method in terms of retrieval accuracy which is evident from the results that we obtain using our technique. The detailed results and discussion is given in the next section.

**Results and Discussion**

The proposed mathematical expression retrieval process is implemented in the working platform of MATLAB. The suggested system has been evaluated with different query images and suitable images are recovered from the image database. The method is based on the feature extraction and next further optimization technique applying the evolutionary programming. The query image must be preprocessed to recover the image from the image database, to accurate the intensity levels of the input image with the intensity levels of the images in the database.

The databases used in the proposed method of retrieval are grouped into two like complex and simple expression data.

\[ x \land (y \lor z) = (x \land y) \lor z \quad A \cap B = A \cup B \]

\[ (A \land B)^c = A^c \cap B^c \quad A \cup B = A \cap B \]

\[ x \lor y = y \lor x \quad x \land y = y \land x \]

**Figure 2:** Simple expression database

\[ \alpha = (K + \sigma^2 I)^{-1} Y \quad \mu(x^*) = \frac{1}{N} \sum_{x_i} \delta(x^*, x_i) \]

\[ Y = \{ y_1, \ldots, y_2 \}^T \quad \delta(x_i - x^*_j)^2 \sigma^2 \]

\[ u_i = \sum_{j=1}^{N} \delta(j) \sum_{j=1}^{N} \sigma^2 \]

\[ \alpha_i = \frac{1}{\sqrt{\alpha_i}} V_i \]

**Figure 3:** Complex database
**Performance Evaluation**

The performance evaluation of the proposed methodology is calculated by measuring the accuracy, sensitivity and specificity of the method. The sensitivity, specificity and accuracy values are calculated using the expressions given below.

\[
\text{Sensitivity} = \frac{TP}{TP + FN} \quad (13)
\]

\[
\text{Specificity} = \frac{TN}{TN + FP} \quad (14)
\]

\[
\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} \quad (15)
\]

Where,

- True positive (TP) is the number of images that are correctly classified.
- True negative (TN) is the number of irrelevant images that are correctly classified.
- False positive (FP) is the number of relevant images that are incorrectly classified as irrelevant images.
- False negative (FN) is the number of irrelevant images that are incorrectly classified as relevant images.

The table below shows the accuracy, sensitivity and specificity values obtained using the proposed method.

**Table 1:** Sensitivity, Specificity and Accuracy for number of input images

<table>
<thead>
<tr>
<th>No of images</th>
<th>Sensitivity</th>
<th>Specificity</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>60</td>
<td>48.57</td>
<td>37.37</td>
</tr>
<tr>
<td>10</td>
<td>66.67</td>
<td>49.31</td>
<td>48.05</td>
</tr>
<tr>
<td>15</td>
<td>71.43</td>
<td>49.55</td>
<td>48.28</td>
</tr>
<tr>
<td>20</td>
<td>75</td>
<td>49.66</td>
<td>48.39</td>
</tr>
<tr>
<td>25</td>
<td>77.78</td>
<td>49.73</td>
<td>48.45</td>
</tr>
</tbody>
</table>

The F-measure for the proposed method is then calculated using the expression,

\[
F = 2 \left( \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \right) \quad (18)
\]

The average F-measure value for the proposed and existing method is found out and the corresponding graph is shown in fig 5.

**Table 3:** Average F-measure for proposed and existing methods

<table>
<thead>
<tr>
<th>F-measure</th>
<th>Methods</th>
<th>Proposed Method</th>
<th>Existing Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8205</td>
<td>Existing [12]</td>
<td>0.60</td>
<td></td>
</tr>
</tbody>
</table>

By applying precision and recall, the performance of the suggested method can be recognized. Precision is the fraction of recovered images that are related to the query image, while recall is the fraction of related images that are recovered from the database. Both precision and recall are therefore based on a perceptual and measure of relevance.
Conclusion
In this paper, we have proposed an effective and efficient mathematical expression retrieval approach which is prone to be error free. The feature extraction is performed for each image in the database and this feature values are used for retrieval of the query images by comparing with the database. The implementation results illustrates that this type of image retrieval process effectively retrieves the images that are very close to the query image from the database when compared to the various existing methods. This could be visualized from the precision and recall plot, determined from the retrieval results.

References


