Integration of Histogram Insertion Measure Method for Achieve Effecient Retreival of Images

Mrs. V. Indrani[1], Mrs. K. Prathima[2], Mr. B. Vikas[3], Mrs. Ch. Roopasree[4]

(Dept of CSE, Vignan's Institute Management and Technology for Women, Hyderabad, Andhra Pradesh, India)

ABSTRACT : *Retrieving information in the form of images had been vide spreading research now a days. Knowledge transmission is achieved through information retrieval in both the forms of Images and Texts, Presently researchers and mainly focusing on this. Mainly in our paper we are going to incorporate the method of Histogram Intersection measure to evaluate the database image to Query image, which helps to measure the over-all similarities between all the images in the Knowledge Database. Hence we prove that our approach is accurate by incorporating the all the local properties of the texture histogram of the images.*

Keywords: Image Retrieval, Texture Features, Knowledge Transmission, Query image, Local Properties

I. INTRODUCTION

To date, image and video storage and retrieval systems have typically relied on human supplied textual annotations to enable indexing and searches. The text-based indexes for large image and video archives are time consuming to create. They necessitate that each image and video scene is analyzed manually by a domain expert so the contents can be described textually. The language-based descriptions, however, can never capture the visual content sufficiently. For example, a description of the overall semantic content of an image does not include an enumeration of all the objects and their characteristics, which may be of interest later. A content mismatch occurs when the information that the domain expert ascertains from an image differs from the information that the user is interested in. A content mismatch is catastrophic in the sense that little can be done to approximate or recover the omitted annotations. In addition, a language mismatch can occur when the user and the domain expert use different languages or phrases. Because text-based matching provides only hit-or-miss type searching, when the user does not specify the right keywords the desired images are unreachable without examining the entire collection.
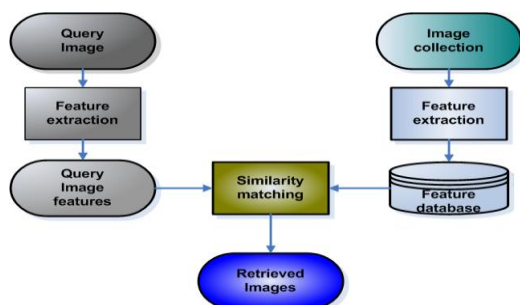

Fig.1: The Architecture of CBIR Technique

The prime requirement for Retrieval systems is to be able to display images relating to a named query image. The text indexing is often limited, tedious and subjective for describing image content. So there is increasing interest in the use of CBIR techniques. The problems with text-based access to images have prompted increasing interest in the development of image based solutions. This is more often referred to as Content Based Image Retrieval (CBIR) as shown in Fig.1. Content Based Image Retrieval relies on the characterization of primitive features such as color, shape and texture that can be automatically extracted from the images themselves. Queries to CBIR system are most often expressed as visual exemplars of the type of the image or image attributed being sought. For Example user may submit a sketch, click on the texture pallet, or select a particular shape of interest. This system then identifies those stored images with a high degree of similarity to the requested feature.

Digital imaging has become the standard for all image acquisition devices. So there is an increasing need for data storage and retrieval. With lakhs of images added to the image database, not many images are annotated with proper description. So many relevant images go unmatched. The most widely accepted content-based image retrieval techniques cannot address the problems with all images, which are highly specialized .Our approach Histogram based Image Retrieval using Texture Feature retrieves the relevant images based on the texture property. We also provide an interface where the user can give a query image as an input. The texture feature is automatically extracted from the query image and is compared to the images in the database retrieving the matching images.

II. BACKGROUND & PREVIOUS WORK

The goal of Content-Based Image Retrieval (CBIR) systems is to operate on collections of images and, in response to visual queries, extract relevant image. The application potential of CBIR for fast and effective image retrieval is enormous, expanding the use of computer technology to a management tool.
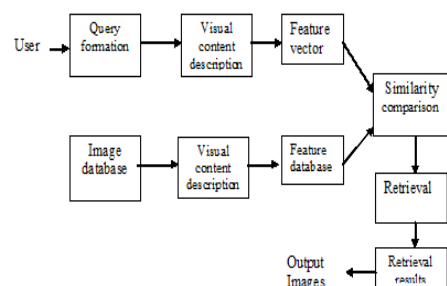

Fig.2: Procedure for content based image retrieval system

CBIR operates on the principle of retrieving stored images from a collection by comparing features automatically extracted from the images themselves. The commonest features used are mathematical measures of color, texture or shape. A typical system allows users to formulate queries

by submitting an example of the type of image being sought, though some offer alternatives such as selection from a palette or sketch input. The system then identifies those stored images whose feature values match those of the query most closely, and displays thumbnails of these images on the screen.
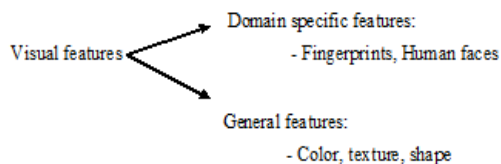

Fig.3: Approaches of CBIR

Some of the most commonly used types of features used for image retrieval are as follows:

a)   Colour retrieval
Several methods for retrieving images on the basis of color similarity have been described in the literature, but most are variations on the same basic idea. Each image added to the collection is analyzed to compute a color histogram, which shows the proportion of pixels of each color within the image. The color histogram for each image is then stores in then stored in the database. At each time, the user can either specify the desired proportion of each color (&75% olive green and 25% red, for example), or submit an example image from which a color histogram is calculated. Either way, the matching process then retrieves those, which a color histogram is calculated. Either way, the matching process then retrieves those images whose color histograms match those of the query most closely.

b)   Texture retrieval
The ability to match on texture similarity can often be useful in distinguishing between areas of images with similar color (such as blue sky and sea or green leaves and grass). A variety of techniques has been used for measuring texture similarity; the best established rely on comparing values of what are known as second-order statistics calculated from query and stored images. Essentially, these calculate the relative brightness of selected pairs of pixels from each image. From these it is possible to calculate measures of image texture such as the degree of contrast, coarseness, directionality and regularity or periodically, directionality and randomness.
Texture queries can be formulated in a similar manner to color queries, by selecting examples of desires texture a palette, or by supplying an example query image. The system then retrieves images with texture measures most similar in value to the query.

c)   Shape retrieval
Two major steps are involves in shape feature extraction. They are object segmentation and shape representation.
Object segmentation: Segmentation is very important to Image Retrieval. Both the shape feature and the layout feature depend on good segmentation allow fast and efficient searching for information of a user's need.

d) Shape representation

In image retrieval, depending on the applications, some requires the shape representation to be invariant to translation, rotation, and scaling. In general, the shape representations can be divided into two categories, boundary-based and region-based. The former uses only the outer boundary of the shape while the latter uses the entire shape region.

III.   CONCEPT OF TEXTURE FEATURE
Texture is one of the crucial primitives in human vision and texture features have been used to identify contents of images. Texture refers to the visual patterns that have properties of homogeneity that do not result from the presence of only a single color or intensity. Texture contains important information about the structural arrangement of surfaces and their relationship to the surrounding environment. One crucial distinction between color and texture features is that color is a point, or pixel, property, whereas texture is a local-neighbourhood property. As a result, it does not make sense to discuss the texture content at pixel level without considering the neighbourhood.
The texture is a property inherent to the surface. Various parameters or textural characteristics describe it. They are:

- The Granularity which can be rough or fine
- The Evenness which can be more or less good
- The Linearity
- The directivity
- The repetitiveness
- The contrast
- The order
- The connectivity

The other characteristics like color, size, and shape also must be considered. The Methodologies used for analysis of the texture are as follows

i.   Texture spectrum method
The basic concept of texture spectrum method was introduced by H1 and Wang. The texture can be extracted from the neighborhood of 3 X 3 window which constitute the smallest unit called 'texture unit'. The neighborhood of 3 X 3 consists of nine elements respectively as $V=\{ V_1, V_2, V_3, V_4, V_0, V_5, V_6, V_7, V_8 \}$ where V_0 is the central pixel value and $V_1....V_8$ are the values of neighboring pixels within the window. The corresponding texture unit for this window is then a set containing eight elements surrounding the central pixel, represented as:

$TU = \{ E_1, E_2, E_3, E_4, E_0, E_5, E_6, E_7, E_8 \}$

Where E_i is defined as: $E_i =$ 0   if $V_i < V_0$

        1   if $V_i = V_0$

        2   if $V_i > V_0$

And the element E_1 occupies the corresponding V_1 pixel. Since each of the eight element of the texture units has any one of three values (0, 1, or 2)

$NTU = \sum E_i * 3^{(i-1)}$ [For i=1 to 8]

Where NTU is the texture unit value. The occurrence distribution of texture unit is called the texture spectrum (TS). Each unit represents the local texture information of 3X3 pixels, and hence statistics of all the texture units in an image represent the complete texture aspect of entire image.

ii.   Cross diagonal texture spectrum

AL-Jan obi (2001) has proposed a cross-diagonal texture matrix technique. In this method the eight neighboring pixels of 3 X 3 widows is broken up into two groups of four elements each at cross and diagonal positions. These groups are named as Cross Texture Unit (CTU) and Diagonal Texture Unit (DTU) respectively. Each of the four elements of these units is assigned a value (0, 1, and 2) depending on the gray level difference of the corresponding pixel with that of the central pixel of 3X3 window. These texture units have values from 0 to 80 (34, i.e 81 possible values).

Cross Texture Unit (CTU) and Diagonal Texture Unit (DTU) can be defined as:

$NCTU = \sum Eci * 3 (i - 1)$ [For i=1 to 4]
$NDTU = \sum Edi * 3 (i - 1)$ [For i=1 to 4]

Where NCTU and NDTU are the cross texture and diagonal texture unit values respectively; Eci and Edi are the ith elements of texture unit

The texture unit (CTU orDTU) value can range between: (0-240)

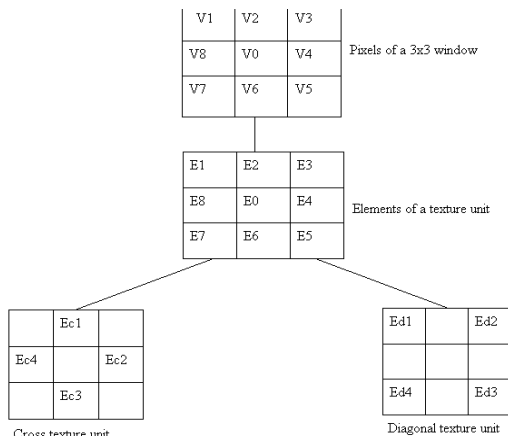*Fig.4: Formation of cross-diagonal texture units*

i.   Modified texture spectrum

In the proposed method, Nctu and Ndtu values have been evaluated which range from 0 to 80. For each type of texture unit, there can be four possible ways of ordering, which give four different values of CTU and DTU.

NTU = NCTU * NDTU
Ntu = Nctu+Ndtu

Ntu = Nctu-Ndtu

Where Nctu and Ndtu are the ordering ways for evaluation of Nctu and Ndtu.

After obtaining the CDTM values of 3*3 windows through entire image the occurrence frequency of each CDTM values are recorded. For the texture units having same CDTM values, two different procedures have been carried out to replace the pixel values of these units. The texture unit value can range between :(0-480).

ii.   Texture spectrum with threshold

The texture spectrum method with threshold is intended to make difference between the values of neighborhood matrix which are very close to the cental pixel value and those the rest.In this method the texture unit matix is represented as:

TU = { E1 , E2 , E3 , E4 , E0 ,E5 , E6 ,E7 , E8 }

Where Ei is defined as:Ei =   0 if $Vi < =(V0 + t)$

1 if $Vi > (V0 + t)$

Where t is the threshold value.

$NTU = \sum Ei * 2 (i - 1)$ [For i=1 to 8]

The texture unit value can range between (0-254).

iii.   Reduced texture unit

In this method the range of texture unit values are (0,1).As the range is decreased the memory required to compute texture unit value also reduces.In this method

TU = { E1 , E2 , E3 , E4 , E0 ,E5 , E6 ,E7 , E8 }

Where Ei is defined as: Ei = 0 if $Vi < = V0$

1 if $Vi > V0$

Where t is the threshold value.

$N RTU = \sum Ei * 2 (i - 1)$ [For i=1 to 8]

The texture unit value can range between (0-254).

iv.   Splitting texture unit matrix into rows and columns

In this approach the texture unit matrix is split into 3 separate rows/columns. Texture unit value is calculated separately for each row/column. Later all the 3 texture unit values are added to get a single texture unit value. By doing this the texture unit value can be limited to 42.Thus memory and computation time can be saved.

Splitting into columns:

Here [E11- E13] are the first column values of the texture unit matrix denoted as TU1. Similarly [E21- E23] and [E31- E33] denote second (TU2) and third (TU3) columns of texture unit matrix respectively.

The texure unit value is calculated separately for each texture unit matrix (j) as:

N TUj = ∑ Eji * 2 (i - 1)   [For i=1 to 3]

The final texture unit value is evaluated as:

N TU = ∑ N TUj   [For j=1 to 3]

The texture unit value can range between (0-42).

Splitting into rows:

E_{11}	E_{12}	E_{13}
E_{21}	E_{22}	E_{23}
E_{31}	E_{32}	E_{33}

Here [E11- E13] are the first row values of the texture unit matrix denoted as TU1. Similarly [E21- E23] and [E31- E33] denote second (TU2) and third (TU3) rows of texture unit matrix respectively.

The texure unit value is calculated separately for each texture unit matrix (j) as:

N TUj = ∑ Eji * 2 (i - 1)   [For i=1 to 3]

The final texture unit value is evaluated as:

N TU = ∑ N TUj   [For j=1 to 3]

The texture unit value can range between (0-42).

IV.   HISTOGRAM INTERSECTION APPROACH

To overcome the disadvantages of Euclidean distance we taken histogram intersection measure. The histogram intersection was investigated for color image retrieval by swain and Ballard. Their objective was to find known objects within images using color histograms. When object (q) size is less than the image (t) size, and the histograms are not normalized, then $|hq|<=|ht|$. The intersection of histograms hq and ht is given by:

$$d_{q,t} = 1 - \frac{\sum_{m=0}^{M-1} \min (h_q[m], h_t[m])}{|h_q|}$$

Where $|h| = \Sigma h[m]$ [for m=0 to M-1].The above equation is not a valid distance metric since it is not symmetric hq,t not equal to dt,q. However that equation can be modified to produce a true distance metric by making it symmetric in hq and ht as follows:

$$d^1_{q,t} = 1 - \frac{\sum_{m=0}^{M-1} \min (h_q[m], h_t[m])}{\min(|h_q|, |h_t|)}$$

Alternatively when the histograms are normalized such that |hq|=|ht|, both equations are true distance metrics.When |hq|=|ht| that D1(q,t)=dq,t and the Histogram Intersection is given by

$$D1_{(q, t)} = \Sigma |h_q[m] - h_t[m]| [\text{For m=0 to M-1}]$$

V.   DESIGN & ANALYSIS OF THE EXPERIMENT

Class Diagram models class structure and contents using design elements such as classes, packages and objects as shown in Fig.5. It also displays relationships such as containment, inheritance, associations and others.
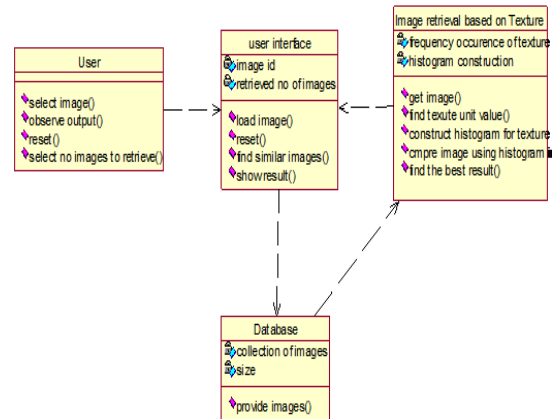

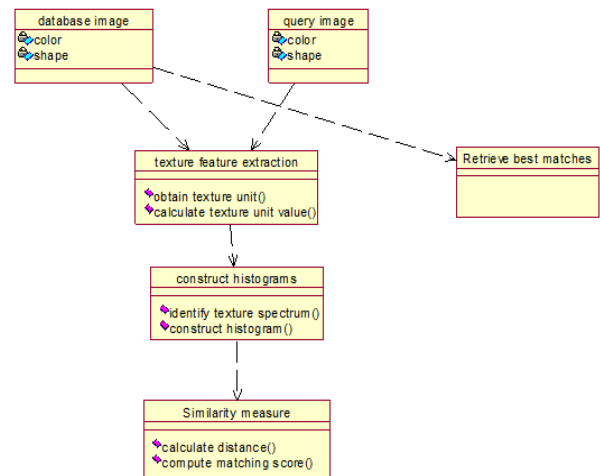*Fig.5: Interpretational Class Diagram for User interaction*

Fig.6: Interpretational Class Diagram for Image Retrieval

Sequence Diagram displays the time sequence of the objects participating in the interaction as shown in Fig.6. This consists of the vertical dimension (time) and horizontal dimension (different objects).
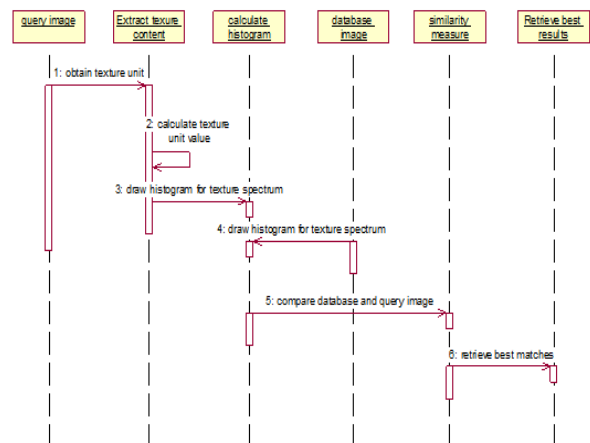

Fig.7: Interpretational Sequence Diagram of Database search

VI. RESULTS
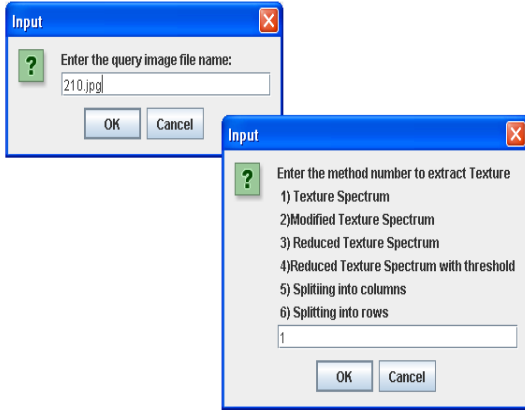
The below are the results obtained from the experiment
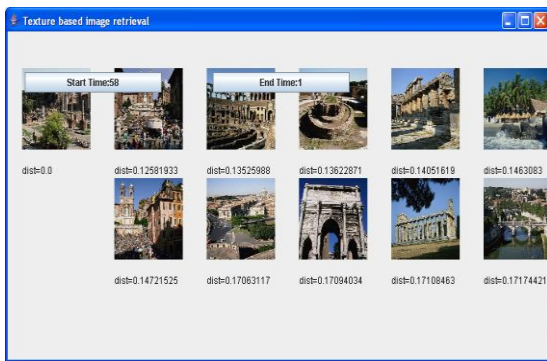

Fig.8: Texture Spectrum Method

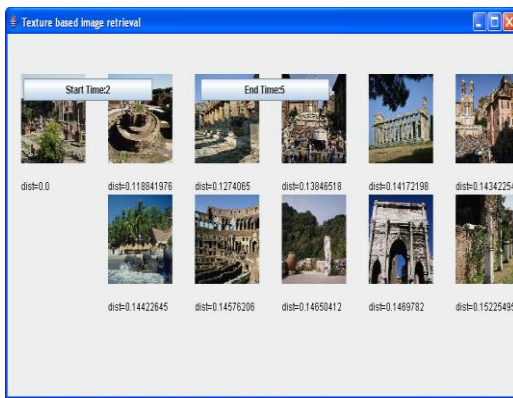*Fig.9: Modified Texture Spectrum Method*

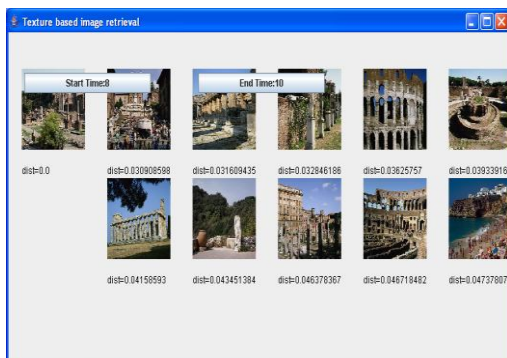*Fig.10: Reduced Texture Spectrum Method*

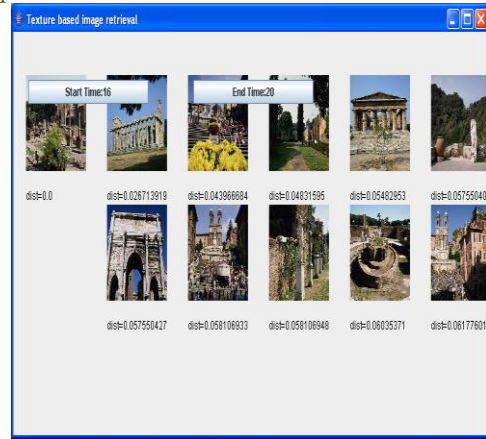*Fig.11: Reduced Texture Spectrum with Threshold*

Fig.12: Splitting into Rows

VII. CONCLUSION

We presented a new idea of achieving Knowledge transmission of image retrieval through Histogram based Image Retrieval using Texture Feature system with different methods of extracting texture feature in both the forms of Images and Texts .We mainly concentrated and successes to incorporate the method of Histogram Intersection measure to evaluate the database image to Query image, which helped to calculate over all similarities between all the images in the Database.

REFERENCES

[1] A. K. Jain and R. Dubes. Algorithms for clustering data. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988

[2] R. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classi.cation. IEEE Transactions Systems on Man and Cybernetics, 3(6):610 621, 1973.

[3] W. Y. Ma and B. S. Manjunath. Netra: a toolbox for navigating large image databases. In International Conference on Image Processing (ICIP), pages 568–571, Oct. 1997.

[4] T. Lehmann, B. Wein, J. Dahmen, J. Bredno,F. Vogelsang, and M. Kohnen. Content-Based Image Retrieval in Medical Applications: A Novel Multi-Step Approach. In International Society for Optical Engineering (SPIE), volume 3972(32), pages 312–320, Feb. 2000.

[5] A. Bovik, M. Clark, and W. Geisler. Multichannel texture analysis using localized spatial .lters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12:55–73, 1990.T. S. Huang, S. Mehrotra, and K. Ramchandran.

[6] F. Korn, N. Sidiropoulos, C. Faloustos, E. Siegel, and Z. Protopapas. Fast and e.ective retrieval of medical tumor shapes. IEEE Transactions on Knowledge and Data Engineering, 10(6):889–904, 1998.

[7] A. Kak and C. Pavlopoulou. Content-Based Image Retrieval from Large Medical Databases. In 3D Data Processing, Visualization, Transmission, Padova, Italy, June 2002.

[8]   W. Chu, C. Hsu, A. Cardenas, and R. T aira. Knowledge-based image retrieval with spatial and temporal constructs. IEEE Transactions on Knowledge and Data Engineering, 10(6):872–888, 1998.

[9]   D. Clausi and M. Jernigan. Designing Gabor filters for optimal texture separability. Pattern Recognition, 33:1835–1849, Jan. 2000.

[10]   J. Bach, C. Fuller, A. Gupta, A. Hampapur,B. Horowitz, R. Humphrey, and R. Jain. Virage image search engine: An open framework for imagemanagement. In SPIE Storage and Retrieval for Imageand Video Databases, volume 2670, pages 76–87, SanDiego/La Jolla, CA, Jan. 1996.

[11]   D. Comaniciu, P. Meer, D. Foran, and E. Medl. Bimodal system for interactive indexing and retrieval of pathology images. In Workshop on Applications of Computer Vision, pages 76–81, Princeton, NJ, Oct.1998.

[12]   Multimedia Analysis and Retrieval System (MARS) project. In Proceedings of 33rd Annual Clinic on Library Application of Data processing - Digital Image Access and Retrieval, pages 100–117, Urbana/Champaign, Illinois, 1996

Author List

Mrs. V. Indrani received M.Tech (CST with Bioinformatics) from Andhra University, Visakhapatnam. Presently she is working as Associate Professor at Vignan Institute Management and Technology for Women, Hyderabad, Andhra Pradesh, India. She is having 7+ years of teaching experience in the field of Computer Science and Engineering.

Mrs. K. Prathima received M.Tech (CSE) from Osmania University, Hyderabad. Presently she is working as Assistant Professor at Vignan's Institute of Management and Technology for Women, Hyderabad, Andhra Pradesh, India. She is having 7+ years of teaching experience in the field of Computer Science and Engineering.

Mr. B. Vikas received M.Tech (Bioinformatics) from JNT University, Hyderabad. Presently he is working as Assistant Professor at Vignan's Institute of Management and Technology for Women, Hyderabad, Andhra Pradesh, India. He is having 2+ years of teaching experience in the field of Computer Science and Engineering.

Mrs. Ch. Roopasree received M.Tech (Bioinformatics) from JNT University, Hyderabad. Presently she is working as Assistant Professor at Vignan's Institute of Management and Technology for Women, Hyderabad, Andhra Pradesh, India. She is having 3+ years of teaching experience in the field of Computer Science and Engineering.