1. INTRODUCTION

The interpretation of image contents has significant role in computer vision and image processing. An image may contain different type of information of a scene, such as object’s shape, color, size and orientation. But separation of the objects from their background is one of the essential tasks that should be performed before any interpretation. In case of contour extraction of an object, we must detect the edge which builds that object, and this fact disclose us the intrinsic importance of edge detection in computer vision and image processing. Edge detection named for the process of identifying and spotting local changes or discontinuities in an image. Edge detection is a vital step in the image processing areas such as image enhancement, object recognition, morphing, image restoration, watermarking, pattern recognition image registration, compression, and image retrieval. Edges characterize the boundaries between regions in an image, which helps in image processing applications like segmentation and object recognition. The image edges include relevant information that is useful for obtaining the image characteristics. Edge detection mentioned as the process of locating points in a digital image at which intensity of image change occurs and so they are characterized into set of curved line segments called as edges. So, edge detection is a fundamental step in image analysis and it forms the basis for solving numerous complex problems.

Edge detection also reduces the amount of data to be represented while retaining significant information about the shape of object and so fundamental of low-level image processing. In the same way, good edges are necessary for higher level processing like pattern recognition. The main objective of good edge detection method is to minimize the probability of detecting spurious edges caused by noise as well as missing real edges. Edge detection has been a challenging problem if the image is affected by noise. Similarly it becomes more challenging when considering color image due to its multi dimensional nature. Color image provides accurate information about objects of a scene than the gray scale images.

There are many ways to perform edge detection. However most of these methods are mainly grouped into two categories namely Gradient based edge detection and Laplacian based edge detection [2]. The gradient based methods detect the edges by computing edge strength (related to the local image contrast along the normal) and searching in a gradient direction or in other words we can say it detect edges by looking local maxima and minima in the first derivative of image. The Laplacian method searches for zero crossings in the second derivative of the image in order to find edges. Edge detection algorithms such as Prewitt, Robert and Sobel are based on gradient based algorithm. In these algorithms, if calculated gradient pixel value higher than a predefined threshold is considered as an edge pixel. Because threshold value is often analytically determined, it is possible to lose some real edges also the occurrence of false edges. Another gradient based edge detection algorithm is Canny which solves an optimization problem to detect the edges [2]. The main problem with this type of classical edge detection approach is that a low threshold produces false edges (spurious edges), but a high threshold misses real edges.

There are many other techniques used for edge detection; some of them are based on fuzzy logic, wavelet approach, morphology, genetic algorithms, neural network and Bayesian approach. Dong Hu and Zianzhong Tian[3] in their paper introduce image fuzzy characteristic into mathematical morphology and then detect the edges using mathematical morphology. Various wavelet based methods[4] namely Discrete Cosine Transform[5], has been also used for successful edge detection. Several neural networks methods[6] have also been introduced for edge detection.
The purpose of this paper is to figure out the differences with Ricci curvature in the context of edge detection. This paper is organized as follows: section 1 is the introduction. Section 2 introduces edge detection using Ricci curvature and their computing approach. Section 3 explains results and discussions of this experiment. Section 4 provides visual comparison of results of Ricci edge detection method with Canny edge detector. Section 5 gives summary of this paper.

2. EDGE DETECTION THROUGH RICCI CURVATURE

Differential geometry is a rich area that includes the study of differential calculus, integral calculus and linear algebra to solve the problems in geometry. In addition it belongs to an area in which it is difficult to make progress. This problem mainly due to the difficulty caused by calculation of metrics on general manifolds, which is necessary for the exploration of properties such as curvature. Curvature analysis is of prominent importance in Image Processing areas such as Computer Graphics and Computer Vision.

One curvature function that is extensively used in differential geometry is Ricci curvature. Ricci curvature measures the deviation of a manifold from being locally Euclidean in various tangential directions. In this paper we rely on the concept of combinatorial Ricci curvature which was previously introduced by Forman in his work [6].

In this section we layout our proposed implementation of the combinatorial Ricci curvature for digital images by following the work of Forman. Ricci curvature function in [6] is defined in the context of cell complexes. Grid, mesh or triangulations are the examples of cell complexes. Here we are taken the grid representation of images.

I. CELL DECOMPOSITION OF IMAGES

Cellular decomposition is necessary (see fig 1) when we represent images as manifolds. A binary image is a combination of black and white pixels. So we can think of pixels as tiles.

![Figure 1: cell decomposition of image](image_url)

For various mathematical purposes, we can change the shape of these tiles. We may use triangles, hexagons, etc instead of squares. When we represent images as mesh, we make use of triangles as basic unit. In fact, the shapes may vary within the same image. The shapes of cells may even be curved but the approach remains the same. Generally, these tiles are called cells.

So, given a 2-dimensional grid, we define cells for all integers x, y as follows (see Fig 2):

- a vertex is \{x\}×{y}.
- an edge is \{x\}×[y,y+1] or [x,x+1]×{y}.

![Figure 2](image_url)

From this we have uniform terminology for all dimensions

- a vertex is a 0-cell,
- an edge is a 1-cell,
- a pixel is a 2-cell,

Consider the cell decomposition of the pixel. So in this every edge and vertices are shared with adjacent pixels.
The result of cell decomposition is what we call as cell complex. One advantage of this approach is that it is unique throughout all dimensions: The boundary of an edge (1-cell) consists of its two end-points called 0-cells (vertex). The boundary a pixel (2-cell) consists of its four edges.

As the image is made of pixels, each pixel is attached to each other along the edges they share. In addition we have: Two adjacent edges (1-cells) share a vertex (0-cells) and two adjacent pixels (2-cells) share an edge.

Hence image decomposition in any dimension stated as: the image is composed of cells in such a way that p-cells are attached to each along (p-1)-cells.

II. COMPUTATION OF COMBINATORIAL RICCI CURVATURE

Referring to the cellular structure, 2-cells are actually the pixels of image and 1-cells are the edge (see Figure 1). It can be either horizontal edge or vertical edge between adjacent pixels. Then combinatorial Ricci curvature for 2D images is:

\[
Ric(e) = w(e) \left( \frac{w(G(0))}{w(G(1))} - \frac{\sqrt{w(G(0))w(G(1))}}{w(G(0)) + w(G(1))} \right)
\]

\[
Ric(e) = w(e) \left( \frac{w(G(0))}{w(G(1))} - \frac{\sqrt{w(G(0))w(G(1))}}{w(G(0)) + w(G(1))} \right)
\]

\[\text{(1)}\]

\[\text{Ric}(e_0)\] denotes the Ricci curvature function along an edge \(e_0\) which is shown in blue edge in Fig 1. For computing \(\text{Ric}(e_0)\) we are considering two parallel edges \(e_1\) and \(e_2\) which is shown as green edges in Fig 1. In Eq. (1) \(w\) denotes the weight functions. These functions are intended to catch geometric entities such as length and area. These weight functions can be found out using two weighting methods namely combinatorial weighting and geometric weights. Our proposed method makes use of combinatorial weighting as weight functions to compute Ricci curvature. Given an edge \(e\), the Ricci curvature \(\text{Ric}(e)\) symbolizes generalized mean of the weights of the cells parallel to \(e\). This type of Ricci curvature function is efficient for detecting edges in horizontal direction and ridges in vertical direction. According to combinatorial weighting method, the weight (see Eq. (1)) of a 2-cell surface element has the corresponding pixel value. The adjacent pixel difference along an edge \(e\) is taken as \(w(e)\).

III. ALGORITHM

1. The input digital image should be separated to RGB plane (three channels) for applying Ricci curvature function separately through each channels.
2. Compute the values of \(w(c)\) and \(w(e)\) using the combinatorial weighting method.
3. Determine \(\text{Ric}(e)\) by taking horizontal edges using Eq. (1).
4. Similarly compute \(\text{Ric}(e)\) by taking vertical edges using Eq. (1).
5. Then average above two sectional curvature to obtain the Ricci curvature for the 2D image.

3. RESULTS AND DISCUSSIONS

In this section describes the results of combinatorial Ricci curvature applied on the 2D images. Taking the concept of cell complexes we applied Ricci curvature function on digital image using combinatorial weighting method. Figure 4 shows our input image. Figure 5, 6 shows an example of Ricci curvature achieved through combinatorial weighting method.
4. VISUAL COMPARISON OF RICCI EDGE DETECTION WITH CANNY EDGE DETECTION

From the literature review of various edge detection methods Canny is the best among available. So this can be used as a basis for visual comparison. In this paper we tested edge detection on various digital images through our proposed method. Here colour images are used for edge detection purposes and thereby it tries to overcome its weakness by providing more information in the form of three colour channels. The algorithm that we applied for canny edge detection is given below:

1. Read in a color image and divide it into its three separate color channels (RGB plane).
2. Run each color channel through the canny edge detector separately to find a resulting colored edge map.
3. Combine the resulting edge maps from each of the three color channels into one complete edge map.

So if there was an edge in any of the three colored edge maps, we added it to the general edge map.

Figure 7 shows canny edge detection performed on the image in Fig 4. By visually itself we could
analyse that Ricci curvature function performs better than the canny. The diagonal lines of input image are not detected in canny method. But it is clearly visible in Ricci edge detection method.

The Ricci curvature function is applied on colour images and its results are shown in Fig.8 and 9. Here, the three colour edges are registered in a single frame.

Figure 8: (a) original image (b) edge detection with canny (c) edge detection with Ricci method
In figure 10, some shadows present in the corner of image lead to spurious edge detection in canny but the same is not in Ricci. So Ricci method avoids false edge detection. So from the experiment results more realistic detection of object’s contour is obtained with combinatorial Ricci curvature.

5. CONCLUSION

Edge detection is the primary step for some of the computer vision applications. The edges provide significant visual information corresponding to geometrical or photometrical variations. It results a set of connected curves after applying an edge detector to an image that indicate boundaries of objects. According to good edge detection methods, resultant edges should be thin as possible. In this paper we have applied Ricci curvature function to few colour images. It shows that the features of the focussed areas are prominent and defocused areas and smooth portions get eliminated totally. Also that the hazy edges are strengthened by different colour contours. Our experimental results show the edge detection capability of Combinatorial Ricci curvature. In some way our test image proposed method shows better results than Canny, and is also suitable in generating low resolution colour images.

6. ACKNOWLEDGEMENT:

The first author wishes to thank Dr.P.Geetha who is a member of review panel who gave some constructive suggestions.
REFERENCES


***