Robust Skeletonization using Hough Transform and Geometric Constraints

Subrata Datta
Faculty of Information Technology
Institute of Technology & Marine Engineering
Diamond Hourbar, West Bengal, India.

ABSTRACT

The skeleton is a continuous planar shape for representation as a kind of primitive of the original. The skeleton efficiently concentrates the topological information of the original shape. It is particularly useful for representing amorphous, irregular shapes that cannot be treated by more conventional geometrical methods. The possible applications include the creation of shape primitives, curve segmentation, logging deformation history of deformable objects as well as image preprocessing for shape recognition. In this paper, we proposed a new method of skeletonization using Hough Transform and geometric constraints. We at first identify all the true and spurious skeleton branches using Hough transform and then eliminate spurious branches using two geometric constraints. The geometric constraints in our case are (i) Ratio of length between main skeleton branch & each sub-skeleton branch (ii) Angle between main skeleton branch & each sub-skeleton branch. Our experiment results are more efficient than existing works.

Keywords- Digital Image Processing, Skeletonization, Hough Transform and Shape Matching.

1. INTRODUCTION

Skeletonization is a widely used transformation in the fields of image analysis and shape recognition. Skeletonization in the plane denotes a process which transforms a 2-D object into a 1-D line representation. The concept of skeleton was introduced by Blum in 1961, under the name of medial axis transform. The definition proposed by Blum corresponds to the intuitive skeleton notion, a sort of minimal representation of a set X under the form of lines of unit width. According to Blum every skeleton point is linked to boundary points that are tangential to its maximal circle. A more formal definition of skeleton was proposed by Calabi in 1966 and is based on the concept of maximal ball. However a skeleton is a set of curves that approximates the local symmetry axis of the shape. Several definitions of skeletons have been proposed in the literature. One of the first was based on a "grass fire" model, i.e., a moving wave front generated by an inward motion of an outline curve with constant speed along a normal vector at every point on the curve. The skeleton is the set of points at which the wave front crosses itself. Blum (1967) suggested a transformation to elicit the skeleton that has become known as the grassfire transformation. The idea can be sketched as follows: let us imagine that 'a fire is lit' along the boundaries of the shape and monitor how it spreads inwards. By taking the set of points where at least two fire-fronts meet, one obtains a connected pattern, formed of line-segments and arcs. If the fire-fronts propagate isotropically (and some general conditions are satisfied, see, e.g., Serra, 1988), this pattern coincides with the skeleton defined by means of maximal circles1. Since the appearance of Blum's paper much effort has been devoted to develop algorithms for the calculation of the skeleton. *Boundary based methods* use a representation of the object's boundary as input and make analytical considerations to obtain a representation of the skeleton. Some algorithms calculate the skeleton directly, while others exploit the fact that for polygonal shapes the skeleton is a sub-graph of the Voronoi diagram (Kirkpatrick, 1979). If the boundary is replaced with a point set obtained by discrete sampling, then the skeleton of the shape can be approximated by calculating the discrete Voronoi diagram of the point set and extracting the skeleton from it (Schmitt, 1989; Brandt and Algazi, 1992). Another possibility is to use a polygonal (or spline) approximation of the boundary instead of discrete sampling [1] and [3].

Any skeletonization algorithm is very sensitive to the noise present in the image and it ultimately restricts the generalization capability of the method. The effectiveness of a skeletonization algorithm is heavily dependent on the pruning capability of the algorithm. So a skeletonization algorithm should consist of the effective pruning method also. Due to the noise present in the image, if we apply a skeletonization method, it will generate so many unwanted or spurious skeleton branches. So until we do not remove the spurious branches from the primary skeleton figure, we shall not get the robust skeleton of the object image. If we are unable to find out the noise-resistant skeleton, we cannot preserve the topology of the object. It is the biggest drawback of any skeletonization techniques. This limitation is the principal motivation behind our present research endeavor of incorporating some geometrical constraints in pruning method. So our main objective is to build up a robust skeletonization method. Our primary aim is to modify the present skeletonization methods using Hough transform. We at first identify all the true and spurious skeleton branches using Hough transform and then eliminate spurious branches using two geometric constraints. The geometric constraints in our case are (i) Ratio of length between main skeleton branch & each sub-skeleton branch (ii) Angle between main skeleton branch & each sub-skeleton branch.

2. BACKGROUND

The skeleton is a useful shape representation for many applications in computer vision and computer graphics. The primary drawback of skeleton is that it is very sensitive to minor perturbations in the boundary of the object, which may occur due to various factors such as discretization, segmentation error, image noise, and so forth. The goal of most skeleton pruning techniques is the removal of spurious branches, resulting typically in a much cleaner and more usable skeleton. In addition, the denoised skeleton can then be used to reconstruct a smoother version of the original version.

Most current skeleton pruning algorithms suffer from the problem that when excess branches are removed, other branches that correspond to fine but perceptually significant features of the object are excessively shortened. This is primarily due to the fact that most pruning methods use a global significance measure to discern between data and noise. Unfortunately, for most measures there is a significant overlap between what is considered noise and data, and when the noise is removed some data is taken with it [2].

In 1962 Paul Hough developed a new transformation technique which is later named Hough Transform (HT), as an honor of the scholar. It is a powerful global method for detecting edges. It transforms between the Cartesian space and a parameter space in which a straight line (or other boundary formulation) can be defined [3] and [5].

2.1 Length of the Skeleton Branches

Image skeletonization is very sensitive to noise. Due to the noise present in the image, many unwanted or spurious skeleton branches are also detected with the actual skeleton. After several experiments, it has been noticed that most of the spurious branches have short lengths in comparison to the longest skeleton branch. From this observation we have set up a threshold value relative to the branch length. All the branches which lie under this threshold value are pruned to get the actual skeleton. Thus this geometric constraint is very much important for skeleton pruning. Though most of the spurious branches are shorten length, yet a certain number of spurious branches have long length. So these long spurious branches cannot be pruned by the threshold value regarding to the branch length. At this point we have applied a new technique.

We find out the angle between every skeleton branch and the longest skeleton branch i.e., the main skeleton branch. This is called the orientation of skeleton branches with the main skeleton branch. So we have set up an appropriate range of orientation of the branches. The branches that do not lie within this range are pruned to get the actual image. So this geometric constraint is also important for skeleton pruning method [1] and [5].

2.2 Some Morphological Operations

The word morphology commonly denotes a branch of biology that deals with the form and structure of animals and plants. We use the same word here in the context of mathematical morphology as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hull. The basic operations are shift-invariant (translation invariant) operators strongly related to Minkowski addition. Let *E* be a Euclidean space or an integer grid, and *A* is a binary image in *E*.

- **Erosion**

The erosion of the binary image A by the structuring element B is defined by:

$A \ominus B = \{ z \in E \mid B_z \subseteq A \}$

where Bz is the translation of B by the vector z, i.e., $B_z = \{ b + z \mid b \in B \} \; \forall z \in E$ When the structuring element B has a center (e.g., B is a disk or a square), and this center is located on the origin of E, then the erosion of A by B can be understood as the locus of points reached by the center of B when B moves inside A. The erosion of A by B is also given by the expression:

$$\mathbf{A \ominus B = \cap_{b \in B} A_z}$$

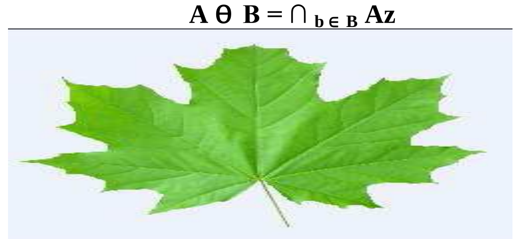


Fig 1 (a): Original RGB image

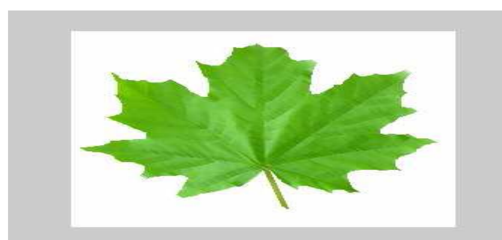


Fig 1(b): Eroded image of fig 1(a)

2.3 The Hit-or-Miss Transformation

The morphological hit-or-miss transform is a basic tool for shape detection [2], [12]. In mathematical morphology, hit-or-miss transform is an operation that detects a given configuration (or pattern) in a binary image, using the morphological erosion operator and a pair of disjoint structuring elements. The result of the hit-or-miss transform is the set of positions, where the first structuring element fits in the foreground of the input image, and the second structuring element misses it completely.

In binary morphology, an image is viewed as a subset of a Euclidean space R^d or the integer grid Z^d, for some dimension *d*. Let us denote this space or grid by *E*. A structuring element is a simple, pre-defined shape,

represented as a binary image, used to probe another binary image, in morphological operations such as erosion, dilation, opening, and closing.
Erosion and dilation can be used in a variety of ways, in parallel and series, to give other transformations including thickening, thinning, skeletonization and many others. Two very important transformations are opening and closing. Now intuitively, dilation expands an image object and erosion shrinks it. Opening generally smoothes a contour in an image, breaks narrow isthmuses and eliminates thin protrusions. Closing tends to narrow smooth sections of contours, fusing narrow breaks and long thin gulfs, eliminating small holes, and filling gaps in contours [6] and [7].

2.4 Basic Morphological Algorithms

When dealing with binary images, one of the principal applications of morphology is in extracting image components that are useful in the representation and description of shape. The basic morphological algorithms are as follows.

- **Boundary Extraction**

The boundary of a set A, denoted by β (A), can be obtained by first eroding A by B and then performing the set difference between A and its erosion [2]. That is, β (A) = A – (A Θ B) where B is a suitable structuring element.

➔Hole Filling

A hole may be defined as a background region surrounded by a connected border of foreground pixels [2]. Let A denote a set whose elements are 8-connected boundaries, each boundary enclosing a background region (i.e. a hole). The objective is to fill all the holes with 1s. We begin by forming an array, X0 of 0s (the same size as the array containing A), except at the locations in X0 corresponding to the given point in each hole, which we set to 1. Then the following procedure fills all the holes with 1s:
Xk = (Xk-1 XOR B) ∩ Ac k = 1, 2, 3, ……..; where B is the symmetric structuring element. The algorithm terminates at iteration step k if Xk = Xk-1. The set Xk then contains all the filled holes. The set union of Xk and A contains all the filled holes and their boundaries.

- **Extraction of Connected Components**

Extraction of connected components from a binary image is central to many automated image analysis applications. Let A be a set containing one or more connected components, and form an array X0 (of the same size as the array containing A) whose elements are 0s (background values), except at each location known to correspond to a point in each connected component in A, which we set to 1 (foreground value) [2]. The objective is to start with X0 and find all the connected components. The following iterative procedure accomplishes this objective:

Xk = (Xk-1 XOR B) ∩A k = 1, 2, 3…; where B is a suitable structuring element. The procedure terminates when Xk = Xk-1 with Xk containing all the connected components of the input image [8].

- **Convex Hull**

A set A is said to be convex if the straight line segment joining any two points in A lies entirely within A [2]. The convex hull of an arbitrary set S is the smallest convex set containing S. The set difference H-S is called the convex deficiency of S. The convex hull and convex deficiency are useful for object description. Here we present a simple morphological algorithm for obtaining the convex hull, C(A), of a set A. Let Bi , i = 1, 2, 3…number of structuring elements. The procedure consists of implementing the equation:

$Xk^{i} = (Xk\text{-}1 \text{ XNOR } B^{i})\text{U } A$, i = 1,2,3,4 (considering 4 structuring element) and k = 1, 2, 3…
With $X0^{i} = A$. When the procedure converges (i.e., when $Xk^{i} = Xk\text{-}1^{i}$), we let $Di = Xk^{i}$. Then the convex hull of A is

$$C(A) = \bigcup_{i=1}^{4} D^{i}$$

- **Thinning**

Thinning is a morphological operation that is used to remove selected foreground pixels from binary images, somewhat like erosion or opening. It can be used for several applications, but is particularly useful for skeletonization. In this mode it is commonly used to tidy up the output of edge detectors by reducing all lines to single pixel thickness. Thinning is normally only applied to binary images, and produces another binary image as output. The thinning operation is related to the hit-and-miss transform, and so it is helpful to have an understanding of that operator before reading on. Like other morphological operators, the behavior of the thinning operation is determined by a structuring element. The binary structuring elements used for thinning are of the extended type described under the hit-and-miss transform (i.e. they can contain both ones and zeros).The thinning operation is related to the hit and miss transform In everyday terms, the thinning operation is calculated by translating the origin of the structuring element to each possible pixel position in the image, and at each such position comparing it with the underlying image pixels. If the foreground and background pixels in the structuring element exactly match foreground and background pixels in the image, then the image pixel underneath the origin of the structuring element is set to background (zero). Otherwise it is left unchanged. Note that the structuring element must always have a one or a blank at its origin if it is to have any effect.

The choice of structuring element determines under what situations a foreground pixel will be set to background, and hence it determines the application for the thinning operation. We have described the effects of a single pass of a thinning operation over the image. In fact, the operator is normally applied repeatedly until it causes no further changes to the image (i.e. until convergence). Alternatively, in some applications, e.g. pruning, the operations may only be applied for a limited number of iterations. Thinning is the dual of thickening, i.e. thickening the foreground is equivalent to thinning the background [9] and [10].

- **Skeletons**

In digital image processing, morphological skeleton is a skeleton (or medial axis) representation of a shape or binary image, computed by means of morphological operators. Morphological skeletons are of two kinds:

(a) Those defined and by means of morphological openings, from which the original shape can be reconstructed,
(b) Those computed by means of the hit-or-miss transform, which preserve the shape's topology. Lantuéjoul's formula.

- **Pruning**

Pruning methods are an essential complement to thinning and skeletonizing algorithms because these procedures tend to leave parasitic components that need to be "cleaned up" by post processing [2]. The skeletons often are characterized by "spurs" (parasitic components). Spurs are caused during erosion by non uniformities in the strokes composing the characters. As a result these parasitic components often change the topology of the object image. It is contradictory to the property of skeletonization. So our duty is to remove these parasitic components to preserve the actual shape of the object. This spurious component removing process is known as pruning.

➔Morphological Reconstruction

The key concept of morphological reconstruction is attached with the concept of geodesic dilation and geodesic erosion [2]. In much of the literature on morphological reconstruction, the structuring element is tacitly assumed to be isotropic and typically is called an elementary isotropic structuring element.

- **Edge Detection**

Edge detection is a fundamental tool in image processing and computer vision, particularly in the areas of feature detection and feature extraction, which aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities.

The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to discontinuities in depth, discontinuities in surface orientation, changes in material properties and variations in scene illumination.

In the ideal case, the result of applying an edge detector to an image may lead to a set of connected curves that indicate the boundaries of objects, the boundaries of surface markings as well as curves that correspond to discontinuities in surface orientation. Thus, applying an edge detection algorithm to an image may significantly reduce the amount of data to be processed and may therefore filter out information that may be regarded as less relevant, while preserving the important structural properties of an image. If the edge detection step is successful, the subsequent task of interpreting the information contents in the original image may therefore be substantially simplified. However, it is not always possible to obtain such ideal edges from real life images of moderate complexity. Edges extracted from non-trivial images are often hampered by fragmentation, meaning that the edge curves are not connected, missing edge segments as well as false edges not corresponding to interesting phenomena in the image – thus complicating the subsequent task of interpreting the image data.

Edge detection is one of the fundamental steps in image processing, image analysis, image pattern recognition, and computer vision techniques. During recent years, however, substantial (and successful) research has also been made on computer vision methods that do not explicitly rely on edge detection as a pre-processing step [1] and [4] and [7].

2.5 Edge Properties

The edges extracted from a two-dimensional image of a three-dimensional scene can be classified as either viewpoint dependent or viewpoint independent. A viewpoint independent edge typically reflects inherent properties of the three-dimensional objects, such as surface markings and surface shape. A viewpoint dependent edge may change as the viewpoint changes, and typically reflects the geometry of the scene, such as objects occluding one another. A typical edge might for instance be the border between a block of red color and a block of yellow. In contrast a line (as can be extracted by a ridge detector) can be a small number of pixels of a different color on an otherwise unchanging background. For a line, there may therefore usually be one edge on each side of the line [11] and [12].

In this paper, we have presented a novel approach for skeleton diagnosis that removes unwanted artifacts while preserving fine features. Here we have used Hough Transform to extract all the skeleton branches. We have also used two main geometric constraints to develop our proposed algorithm for robust skeletonization.

3. PROPOSED SKELETONIZATION METHOD

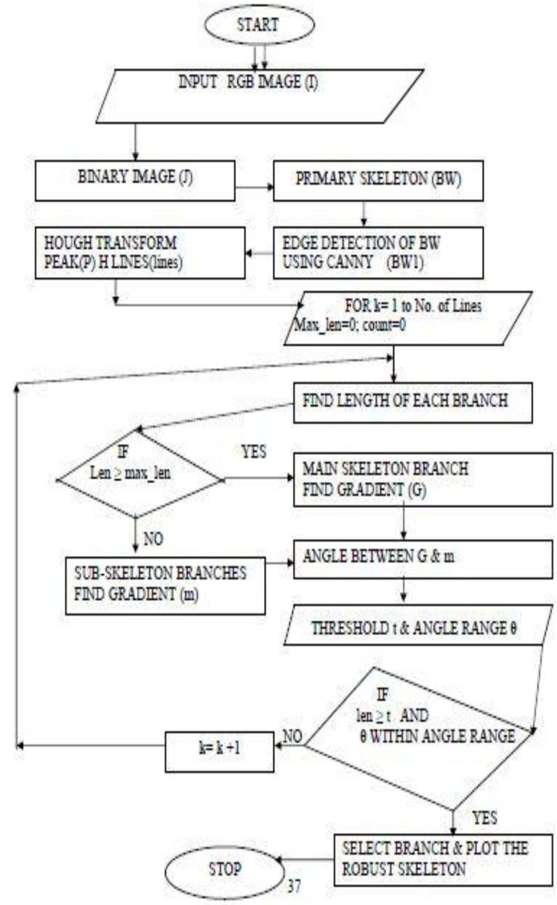
There are many algorithms used for skeletonization. But each of them has some drawbacks. In fact, none of these methods are able to find out the correct skeleton in noisy images. Hence, we propose a new skeletonization method based on Hough transform and two geometric constraints. The two geometric constraints are: (i) Length of the skeleton branches; (ii) Orientation of the skeleton branches (with respect to the longest skeleton branches). The whole process has two iterations. We first prune the noisy skeleton by using the first geometric constraint. We have chosen length as one of the parameters, because most of the spurious skeleton branches have small length. So, we can easily mark those short branches using Hough lines. Then, a threshold value [3], [4], [9] is experimentally chosen and the branches yielding smaller lengths than this threshold are deemed spurious and are hence removed from the image skeleton.

In the second pass, we eliminate comparatively long spurious branches by using the second geometric constraint. Some spurious branches are long too. So, they lie above the threshold value that we have set up for the first iteration. We assume that the branches which are parallel to the longest branch or overlapped with the longest branch are spurious branches. The branches which subtend a small angle with the longest branch are deemed as spurious branches. So, we set up an angle range for pruning the long skeleton branches to get the robust skeleton of the object image.

The pseudo code for the proposed algorithm is presented below:

Input: Binary Image.
Output: Skeleton without spurious branches.
Begin
Find primary skeleton
Detect edge of the skeletal image.
Do the Hough Transform
For k = 1 to number of lines do
Begin
Find length of each skeleton branch
Detect length & gradient of longest line
End For
For k = 1 to number of lines do
Begin
Find gradient of each sub-skeleton branch
Evaluate the angle between sub-skeleton and longest line
End For
For k = 1 to number of lines do
Begin
Set angle range θ
Set threshold t
If angle satisfy θ *and length* $\geq t$
Then prune the spurious branches & plot the robust skeleton
End If
End For
End

3.1 Flowchart of the Proposed Skeletonization Method



3.2 Methodology

We have already presented a new algorithm for finding the robust skeleton of an object image. In the following we are describing the operational procedure in detail. There are basically eight main task of the method. They are:

(1) ***Converting RGB Image into Binary Image:*** The input of the system is RGB image of any type like .PNG, .JPG, .GIF etc. Our first task is to convert this RGB image into binary form. Here we have used MATLAB command 'im2BW' to do the task.

(2) ***Finding Primary (Noisy) Skeleton:*** We applied 'bwmorph' command to find the primary skeleton of the object image. But the output we get at this stage is not robust as because the skeleton have many spurious branches due to the noise present in the image. Our objective is to prune these spurious branches. We then rotate the skeleton image. Primary skeleton contains some unwanted branches in between image skeleton and image frame. To omit these branches we rotate the skeleton image a certain angle.

(3) ***Edge of the Primary Skeleton:*** We have chosen Canny edge detector to find out the edge of the skeleton image.

(4) ***Hough Transform:*** In the next step we employ Hough transform of the skeleton image to find the Hough Peaks &

Hough lines. Basically this method transforms the straight line from Cartesian Co-ordinate plane i.e. (x, y) to a new (m, c) plane. This is also called Rho-Theta plane. The mathematical formula is,

$$\boldsymbol{\rho = x*\cos(\theta) + y*\sin(\theta)}$$

As a result, each of the Hough Peaks may contain more than one straight line passing through a common point in the (x, y) plane. Now we know the gradient (m) and intercepted length along to y axis (c) of every line. By applying the formula of straight line passing through one point (y = mx + c) we find out Hough lines.

(5) ***Selecting Main Skeleton Branch:*** After detecting the Hough lines we find out the coordinates of two terminating points of those lines. Then we find out the length of each line using the following formula.

$$\mathbf{L = [(x2 - x1)2 + (y2 - y1)2]}$$

We identify the branch whose length is the highest and consider that this is the main skeleton branch. The main skeleton branch is marked with color blue. Also we denote the gradient of the main branch as m1 and length L.

(6) ***Finding Angle between Each Sub-skeleton Branch & Main Skeleton Branch:*** To find out the angle between a sub-skeleton branch & the main skeleton branch we need to know the gradient of those branches. We have calculated the gradient of all the sub-skeleton branches. Gradient of each sub-skeleton branch is denoted by m2. Gradient of main skeleton branch is denoted by m1. Then we have applied the following formula to calculate the corresponding angle between the main skeleton branch and one sub-skeleton branch.

$$\mathbf{\tan\theta = |\ (m1-m2)/(1+m1m2)|}$$

This angle measurement will help us to observe the orientation of every branch with respect to the main skeleton branch. This orientation is one of the main parameter in our proposed method.

(7) ***Pruning with respect to Branch Length Threshold Value:*** As mentioned before, the first pruning criterion is based on the comparison of length of sub-skeleton branch & main skeleton branch. We have decided that the sub-skeleton branches whose length is smaller than 35% of the main skeleton branch length will be deemed as spurious branches. So we shall accept only those branches whose lengths are minimum 35 % of that of the main branch. As a result at this a number of spurious branches are removed from the primary skeleton.

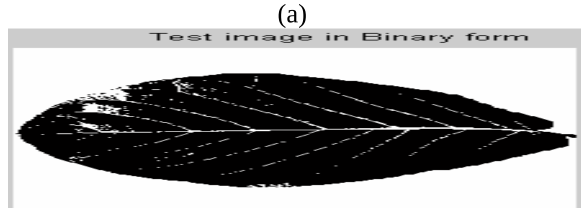
(8) ***Pruning with respect to Branch Orientation Range:*** The second pruning criterion is based on the orientation of the sub-skeleton branches with respect to the main skeleton branch. We have set up an angle range to denote the orientation. The branches, whose orientation angles lie within this range, are treated as the true branches and the remaining are considered as spurious branches. Here we have taken the range as: $0^0 \leq \theta \leq 45^0$.

We have applied two geometric constraints. At first branches are selected according to the threshold value corresponding to branch length. The branches whose lengths are higher than specified threshold value are selected for the next operation. In the second case branches are selected according to the angle range corresponding to the orientation of sub-branches with respect to the main branch. The output of the first case is the input of the second case. Then those branches are selected whose angle with main branch lie within the specified angle range. The output of the second case gives us a better and robust skeleton structure of the input object image.

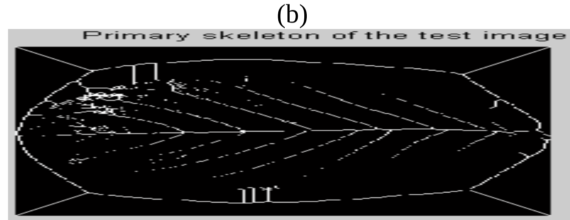
4. RESULTS ANALYSIS

We have chosen leaf database for our testing purpose. The images of vein and sub-veins of a leaf are likely to be straight lines. That's why we have taken leaf database as our sample images. Here in the first case we have taken a lemon leaf picture as the test image. It is a RGB (Red Green Blue) image of type PNG. So, the name of the input image is "lemon.png". The dimensions of the image are 256 X 256 and size is 69.7 kb. Our objective is to find out the robust skeleton of this image. The output images are shown step by step.

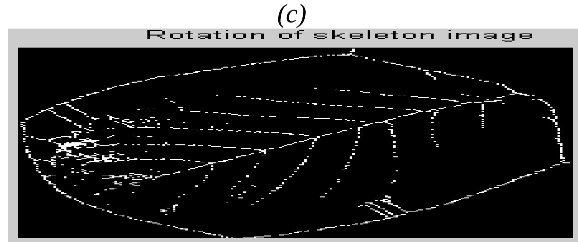
(a)



(b)

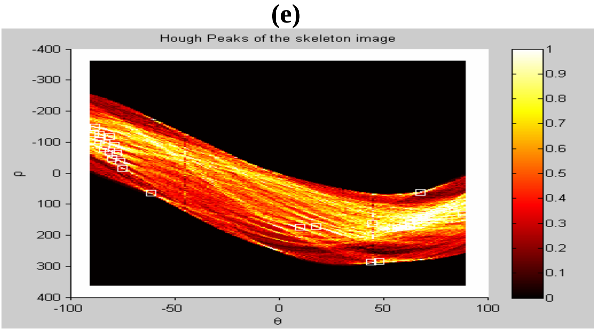


(c)

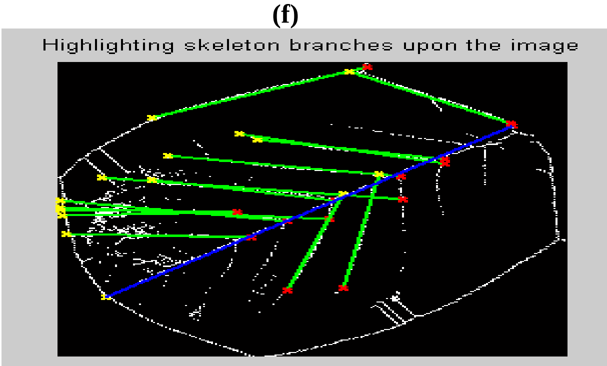


(d)

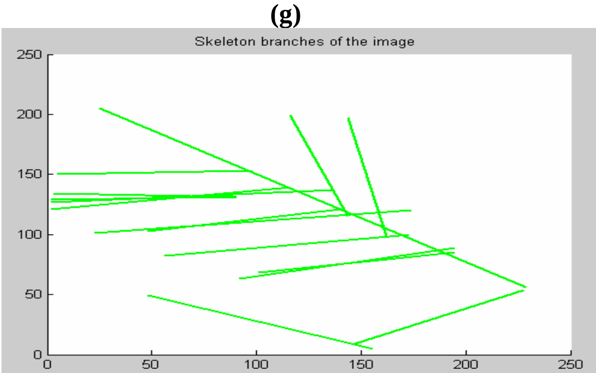
(e)



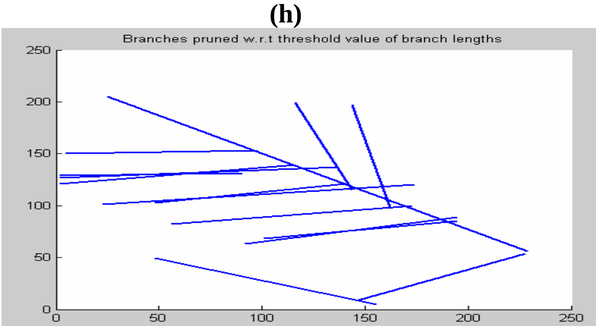
(f)



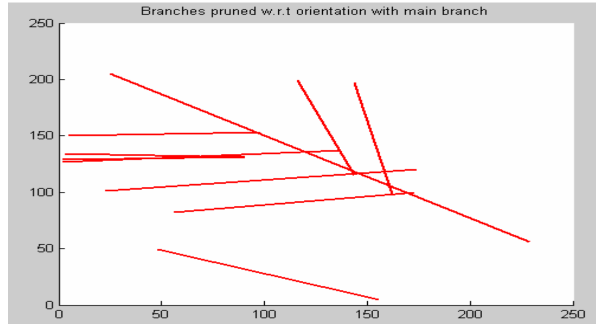
(g)



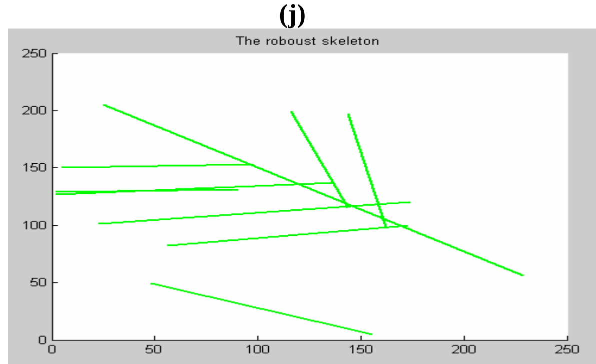
(h)



(i)



(j)



(k)

Figure 2: (a) Test image 'lemon.png' in RGB form.
(b) Test image 'lemon.png' is converted from RGB to Binary form.
(c) Skeleton of test image. This is a noisy skeleton. So we call it primary skeleton.
(d) Rotation of skeleton by 45 degree.
(e) Edge of the rotated skeleton image.
(f) Hough Peaks are plotted in a rho vs theta plane. Peaks are marked with white squares. We have chosen 27 hough peaks.
(g) Corresponding Hough lines of figure 1(f) are aligned with figure 1(d) and highlighted. The longest line is marked with blue and all other lines are marked with green color. The starting & ending positions are marked with yellow & red cross sign respectively.
(h): All the skeleton branches are shown by green straight lines. Here we see that there are total 15 branches in the figure.
(i): The result of the pruning due to application of only first geometric constraint i.e. threshold value of branch length. Here we see that there are total 14
branches in the figure. So (15-14) = 1 branch is pruned.
(j): The result of the pruning due to application of only second geometric constraint i.e. orientation of branch lengths w.r.t the main branch. Here we see that there are total 10 branches in the figure. So (15-10) = 5 branches are pruned.
(k): The robust skeleton due to the application two geometric constraints. Here we see that there are all total 9 branches in the figure. So (15-9) = 6 branches are pruned.

4.1 Statistical Analysis of the Result-

Branch_ number	Co-ordinate (x_1,y_1)	(x_2,y_2)	Length of the line	Gradient of the line (m)	Angle with the longest line (θ)
1	(25, 205)	(229, 56)	252.6203	- 0.7304	0
2	(23, 101)	(174, 120)	152.1907	0.1258	43.3158
3	(56, 82)	(173, 100)	118.3765	0.1538	44.8903
4	(48, 49)	(156, 5)	116.6190	- 0.4074	13.9778
5	(101, 68)	(195, 85)	95.5249	0.1809	46.3953
6	(92, 63)	(195, 89)	106.2309	0.2524	50.3112
7	(2, 129)	(91, 131)	89.6225	0.0225	37.4314
8	(48, 103)	(140, 121)	93.7443	0.1957	47.2143
9	(147, 9)	(228, 54)	92.6607	0.5556	65.1987
10	(2, 127)	(137, 137)	135.3699	0.0741	40.3805
11	(144, 115)	(116, 199)	88.5438	- 3	35.4210
12	(3, 134)	(91, 131)	88.0511	- 0.0341	34.1916
13	(5, 150)	(98, 153)	93.0484	0.0323	37.9917
14	(2, 121)	(116, 139)	115.4123	0.1579	45.1165
15	(162, 98)	(144, 197)	100.6231	- 5.5000	43.5511

TABLE 1: Coordinates of terminating points, length, gradient, angle with the longest line of each skeleton branch.

From TABLE 1 we see that the longest branch i.e. the main branch is Line_number 1 whose length is 252.6203 unit.

***CASE 1:* When we consider only the length threshold value-**
We choose length threshold value t = 35 % of length of the longest line. So the value of t becomes as t = 0.35 x 252.6202 = 88.417105 unit. We have accepted those branches whose length is minimum 88.417105 unit. In TABLE 2 acceptable branch lengths are shown.

Branch_number	Length	Acceptability
1	252.6203	Acceptable
2	152.1907	Acceptable
3	118.3765	Acceptable
4	116.6190	Acceptable
5	95.5249	Acceptable
6	106.2309	Acceptable
7	89.6225	Acceptable
8	93.7443	Acceptable
9	92.6607	Acceptable
10	135.3699	Acceptable
11	88.5438	Acceptable
12	88.0511	Non Acceptable
13	93.0484	Acceptable
14	115.4123	Acceptable
15	100.6231	Acceptable

TABLE 2: Selection of branches whose length is minimum 35 % of longest skeleton branch.

We observe that only one branch is less than the threshold value. So that branch is pruned. So numbers of branches come down from 15 to 14.

***CASE 2:* When we consider only the orientation of branches-**
In the second case we have set up an branch orientation range. We have calculated angle between each sub-branch & the main branch. Then we have decided to accept those branches whose orientation lie within the range $0^0 \leq \theta \leq 45^0$. In TABLE 3, the acceptable branches are shown due to orientation.

Branch number	Angle with the longest line (θ)	Acceptability
1	0	Acceptable
2	43.3158	Acceptable
3	44.8903	Acceptable
4	13.9778	Acceptable
5	46.3953	Non Acceptable
6	50.3112	Non Acceptable
7	37.4314	Acceptable
8	47.2143	Non Acceptable
9	65.1987	Non Acceptable
10	40.3805	Acceptable
11	35.4210	Acceptable
12	34.1916	Acceptable
13	37.9917	Acceptable
14	45.1165	Non Acceptable
15	43.5511	Acceptable

TABLE 3: Selection of branches whose orientation lies within $0^0 \leq \theta \leq 45^0$.

From TABLE 3 we observe that only 10 branches out of 15 are in acceptable range. As a result 5 branches are removed from the primary skeleton.

***CASE 3:* When we consider both of the geometrical constraints-**
When both of the geometrical constraints are implemented on the primary skeleton then we get the robust skeleton structure. This implies that branches will have minimum length 88.417105 unit & will stand within the angle range $0^0 \leq \theta \leq 45^0$. In TABLE 4 all the acceptable branches are shown when both of the constraints are implemented.

Branch_number	Length	Angle with the longest line (θ)	Acceptability
1	**252.6203**	**0**	**Acceptable**
2	**152.1907**	**43.3158**	**Acceptable**
3	**118.3765**	**44.8903**	**Acceptable**
4	**116.6190**	**13.9778**	**Acceptable**
5	95.5249	46.3953	Non Acceptable
6	106.2309	50.3112	Non Acceptable
7	**89.6225**	**37.4314**	**Acceptable**
8	93.7443	47.2143	Non Acceptable
9	92.6607	65.1987	Non Acceptable
10	**135.3699**	**40.3805**	**Acceptable**
11	**88.5438**	**35.4210**	**Acceptable**
12	88.0511	34.1916	Non acceptable
13	**93.0484**	**37.9917**	**Acceptable**
14	115.4123	45.1165	Non acceptable
15	**100.6231**	**43.5511**	**Acceptable**

TABLE 4: Selection of branches when both of the two geometric constraints are considered.

Here we can observe that only 9 branches out of 15 are acceptable one. So we think that these branches are true branches and thus the skeleton structure we get is the robust one.

4.2 Performance Measurement

The performance of skeleton matching depends directly on the property of topology preservation. This means the shape matching. Our proposed method of skeleton pruning minimizes the number of branches of the primary skeleton structure. In TABLE 5 percentage change of number of branches is shown.

Method	Total number branches	Number of branches pruned	% of branches pruned
Only length constraint	15	1	6.25
Only orientation constraint	15	5	33.33
Our proposed method	15	6	40

TABLE 5: Comparison among different methods.

In the following table we are presenting a chart as well as graphical representation of comparison among various outputs obtained after implementing our method.

Input Image	Number of Hough Peaks	Fill Gap	Min Length	Length of Main Skeleton Branch (unit)	Threshold percentage w.r.t main branch	Threshold Value(t)	Range of Angle	Total Number of Branches in the Primary Skeleton	Total Number of Branches Pruned	% of Branches Pruned
lemon.png	27	16	88	252.6203	35 %	88.0511	0-45	15	6	40
coconut.jpeg	27	15	88	270.1426	35%	94.5499	0-45	21	6	28.57
ivy. jpg	15	42	88	306.8697	35%	107.4044	0-45	15	4	26.66

TABLE 6: Comparison among different outputs obtained by implementing our method.

Our proposed method is a new one using Hough transform and geometric constraints. Our method is able to find all the true and spurious branches of the primary (noisy) skeleton. To differentiate between true and spurious branches the method uses two geometric constraints. When the spurious branches are identified then those are removed from the primary skeleton and thus we get a robust skeleton structure of the object image. Our method is a fast and simple one and easy to handle.

5. CONCLUSION AND FUTURE WORKS

Skeletons are useful in the area of Handwritten and printed characters, Fingerprint patterns, Chromosomes and biological cell structures, circuit diagrams, engineering drawings. The objective of our project was to find out a robust skeleton of an object image. To do this we at first have studied the various types of morphological operations related to the shape matching of the object. Secondly we have made an analysis of the characteristics of skeleton of an object image. In this paper, we discussed about the basics of skeletonization, along with its applications. We have also explained the usual pruning techniques. We also showed some advantages and disadvantages of some known skeletonization algorithms. A layout of the paper is presented and also discussed about Hough transform and some geometric constraints. Also a brief description of different morphological operations & edge detection techniques is presented. We proposed a new skeletonization algorithm. The goal was to modify the usual pruning methods for better skeleton extraction of an object image. In particular, we represent the new method with the help of Hough transform and two geometric constraints. This is a new approach of skeletonization using Hough transform.

We plan to extend the proposed method for shape matching. Also this new method can be modified for skeleton graph matching technique. Our skeleton representation technique is based on Hough transform and two geometrical constraints. The new algorithm is limited to two parameters. If more parameters can be added then the output skeleton may be more robust. Beside this we have chosen the primary skeleton branches as straight lines. So here also a modification may be done in future.

6. REFERENCES

[1] Ramesh Jain, Rangachar Kasturi and Brian G. Schunck, "Machine Vision" McGraw-Hill, 1995.

[2] Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing", Peasrson Education, 2008.

[3] Dr. Amit Konar, "Artificial Intelligence and Soft Computing", CRC, 1999. [4] Dr. Amit Konar, "Computational Intelligence-Principles, Techniques and Applications", Springer, 2005.

[5] Xiang Bai, Longin Jan Latecki and Wen-Yu Liu, "Skeleton Pruning by Contour Partitioning with Discrete Curve Evolution", IEEE trans,Pattern Anal. Mach. Intel. 29(3), 2007, 449-462.

[6] H.Blum, "Biological Shape and Visual Science (Part I)", Journal of *Theoretical Biology*, 38:205-287, 1973.

[7] Mrs. J Komala Lakshmi and Dr. M. Punithavalli, "Computation of Merging Points in Skeleton Based Images ", published in *Global Journal of Computer Science and Technology*.

[8] Xiang Bai and Longin Jan Latecki, " Discrete Skeleton Evolution", EMMCVPR, pp. 362- 374, 2007.

[9] Roger Tam and Wolfgang Heidrich, " Feature-Preserving Medial Axis Noise Removal ", LNCS 2351, pp. 672-686, 2002.

[10] R.L. Ogniewicz and O. Kubler, "Hierarchic Voronoi Skeletons", *Pattern Recognition*, 1995.

[11] Polina Golland and W. Eric. L. Grimson, "Fixed Topology Skeletons", CVPR, pp. 10-17.

[12] Luc Vincent, "Efficient Computation of Various Types of Skeletons", SPIE Vol. 1445, 'Medical Imaging V', pp. 297-311, 1991.

7. AUTHOR'S PROFILE

Mr. Subrata Datta presently working as Assistant Professor of Information Technology department at Institute of Technology & Marine Engineering, Diamond Hourbar, West Bengal, India. The degree of B.E. secured in Computer Science & Engineering from SIT, W.B., India, in 2004, and M.Tech. secured in Intelligent Automation & Robotics from Jadavpur University, Kolkata, W.B, India, in 2011. Research Interest includes Artificial Intelligence, Image processing, Machine Vision, and Robotics & Soft-Computing.