ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

Analysis of Convex Hull's Algorithms and It's Application

Rashmi Kulkarni[1], Bhavani K[2]

P.G. Student, Department of Information Science and Engineering, DSCE, Bengaluru, India[1]

Assistant Professor, Department of Information Science and Engineering, DSCE, Bengaluru, India[2]

ABSTRACT: Computational geometry is the branch of computer science [1] where the study of algorithms is done to solve the geometric problems. In such geometric problems the Convex Hull problem is also one among them. In this paper the algorithms of Convex Hull for 2D systems are considered i.e. Graham's Scan algorithm and Jarvis's March algorithm[1]. The time complexity of two algorithms is compared. And the application of Convex Hull in image processing is explained by using MATLAB.

KEYWORDS: algorithms, graham's scan, time complexity, Jarvis's march, image processing

I. INTRODUCTION

Convex Hull problem [1][3][9] is the most broadly studied geometric problem in computational geometry. Many researches have started on efficient convex hull algorithms for the applications in the fields like image processing, computer graphics, robotics, pattern recognition and many more.

Definition –

Let **S** be a set of given points. Convex hull is a polygon, which is the subset (**CH**) of set **S**. In which the points lying on the boundaries are the elements of subset **CH** and points which are lying inside the polygon are the remaining elements of set **S**.Fig.1 is an example of Convex Hull.

$$S=\{p_0,p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9\}$$

And

$$CH=\{p0,p1,p2,p3,p4,p5\}$$

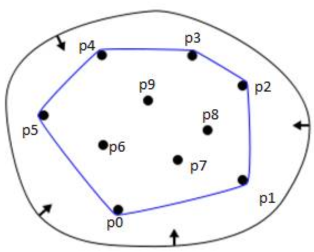

Fig.1 Convex Hull

A convex hull can be described by comparing the set of points to nails hammered into a board. The polygon formation can be seen on the hammered board by wrapping the outer most nails by elastic rubber band.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

II. RELATED WORK

Veljko Petrović et al. [2] has outlined how the temporal complexity of Graham's Scan can be linearized provided it operates on a finite, countable subset of reals that can be represented on some digital computer. Future scope of this was to do the analysis of time complexity of different algorithms for convex hull problem. Jingfan Fan et al. [3] proposed a novel convex hull aided registration method (CHARM) to match two point sets subject to a non-rigid transformation. The proposed algorithm performs several state-of-the-art ones with respect to sampling, rotational angle, and data noise for on both synthetic and real data. The proposed CHARM algorithm also shows higher computational efficiency compared to these methods. GAO Yang et al. [7] proposed a quick convex hull building algorithm using grid and binary tree for the minimum convex building of planar point set. By comparing with the currently representative algorithms of minimum convex hull building, it is found that the proposed algorithm can better describe the profile of irregular objects and its time complexity is relatively low. Artem Potebnia et al. [8] proposed an Innovative algorithm for forming graph minimum convex hulls using the GPU. High speed and linear complexity of this method are achieved by distribution of the graph's vertices into separate units and their filtering. Xujun Zhou et al. [9] Considered the time complexity and newly added samples and proposed an incremental convex hull algorithm based online Support Vector Regression (ICH-OSVR) , which can significantly reduce the time consuming and realize fast online learning when added a new sample. Found that the proposed method can save a lot of time and also more memory. L. Cinque+ et al. [15] proposed a parallel version of the Jarvis' s march, realized using the BSP model and which takes O(nh/p) time (where p is the number of processors and n is the problem size) against the O(nh) complexity of the sequential algorithm. Purpose of this work was to present a very efficient parallel algorithm for computing the convex hull in the plane. Found that the theoretical performance of the algorithm, predicted using the BSP cost formula, closely match the actual running times of this implementation.

In this survey, found that many researchers tried to find algorithms for solving convex hull problem with less time complexity. And many applications are implemented by choosing best algorithm for building convex hull. Here in this work, we tried to analyse the 2 algorithms with their time complexity and tried to choose algorithm which works better with low time complexity. And which algorithm is faster for building convex hull.

III. ALGORITHMS

There are several algorithms to solve the convex hull problem with varying runtimes. Few are mentioned below-

A) Graham's Scan
B) Jarvis's March

A. *Graham's Scan Algorithm*

This algorithm solves the convex hull problem by using a stack S of input points. Let Q be the set of input points. At a time, it pushes all the input points onto the stack, and it pops the points which are not lying on the boundary. So that finally stack S contains the vertices of convex hull CH (Q), in counter clockwise order in the graph. The convex hull is a polygon so minimum 3 points are required to form a polygon therefore input set Q should be $|Q|\geq 3$.The algorithm calls TOP(S) function, which returns the top most element of the stack S. Then it calls NEXT-TO-TOP(S) function, which returns the element down to the top element in the stack. Finally the stack S of Graham's Scan contains the vertices of convex hull from bottom to top, exactly in counterclockwise order.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

GRAHAM-SCAN (Q)
1 let p0 be the point in Q with the minimum y-coordinate.
2 let (p1, p2,….,pm) be the remaining points in Q, sorted by polar angle in counterclockwise order around p0(if more than one point has the same angle, remove all but the one that is farthest from p0)
3 let S be an empty stack
4 PUSH (p0, S)
5 PUSH (p1, S)
6 PUSH (p2, S)
7 for i = 3 **to** m
8 **while** the angle formed by points NEXT-TO-TOP(S), TOP(S), and pi makes a nonleft turn
9 POP(S)
10 PUSH (p_i, S)
11 **return** S

1) *Working of algorithm*

Fig.2 illustrates the working of Graham's scan algorithm [1]. In the above algorithm Step 1, chooses the point p0 as the point with the lowest y-coordinate. Step 2, sorts the remaining points of Q by polar angle with reference to p0, using comparing cross product. Let m be the number of points other than p0 in the set Q. The polar angle is measured in radians. Here the points lay in the 1st and 2nd quadrant means points lie in the half-open interval $[0,\pi)$.

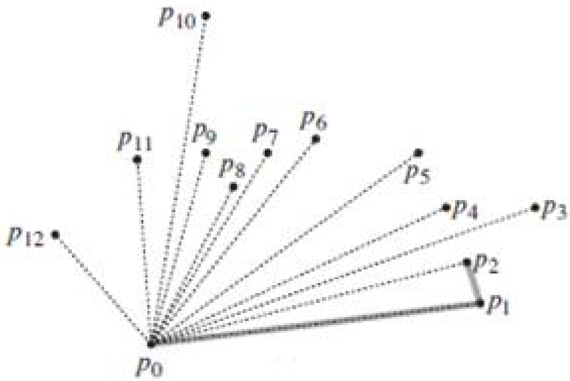

Fig 2.1[1]

The fig 2.1[1] indicates the points of set Q sorted according to their polar angles with reference to p_0 in increasing counterclockwise order. Step 3 initializes the empty stack and pushes the points $p_{0,}$ p1, p2 in the stack. Step 7, the for loop iterates for all the points in the set Q. Step 7,pops the points which makes nonleft turn. That means pops the points which are not going to form the convex hull. Finally it returns the stack of points which forms the convex hull in counterclockwise order.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

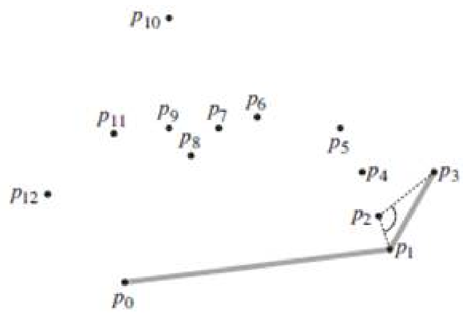

Fig 2.2[1]

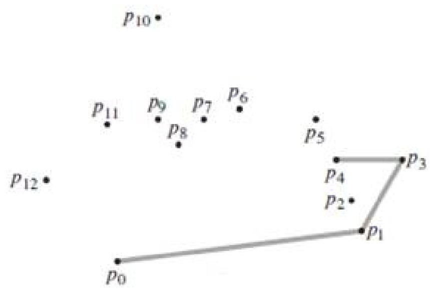

Fig.2.3[1]

Fig 2.2 shows the formation of convex hull, the origin is taken as P0 and from p0 it started tracing in anticlockwise order. P1 , p2 and p3 are pushed on the stack. But p2 to p3 it is making non-left turn and if a triangle is drawn for p0, p1 and p3 , it forms the triangle with p2 inside the triangle. It makes the polygon with more edges which don't want. So pop out the p2 from the stack. In fig.2.3 shows pushing of p4 on the stack . It makes left turn.

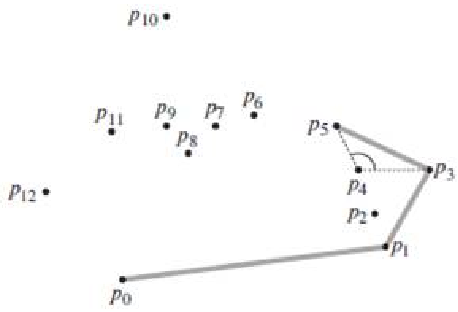

Fig 2.4[1]

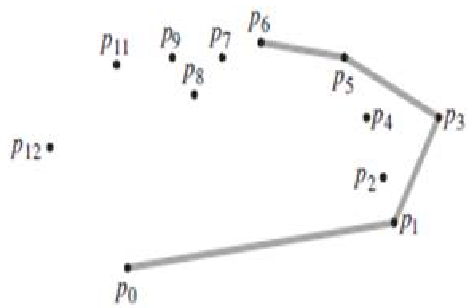

Fig 2.5[1]

But when p5 is pushed on the stack , it makes non-left turn so we have to remove p4 from the stack. Popped p4 to create polygon with less edges. Shown in fig.2.4 . In the fig 2.5 the p6 is pushed on stack, its making left turn from p5. So it should be there in the stack.

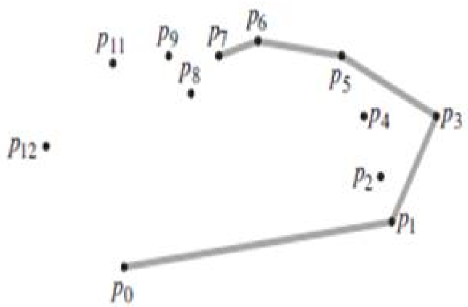

Fig.2.6[1]

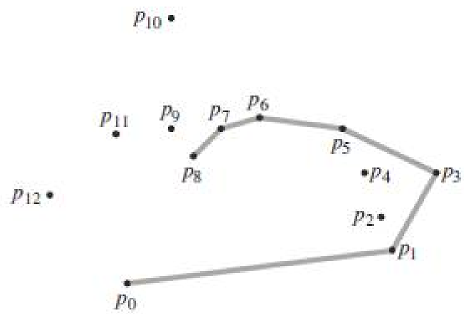

Fig.2.7[1]

In fig 2.6, p7 is pushed on the stack, so it is making left turn from point p6 so it should be there on stack. In fig 2.7, p8 is pushed on to the stack, its making left turn, so it should be there on stack.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

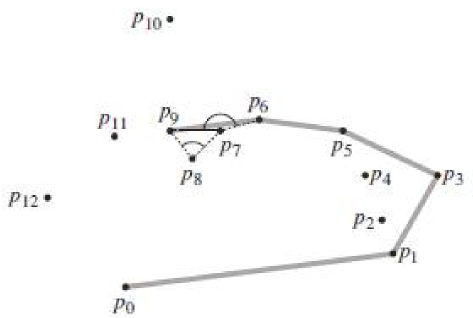

Fig.2.8[1]

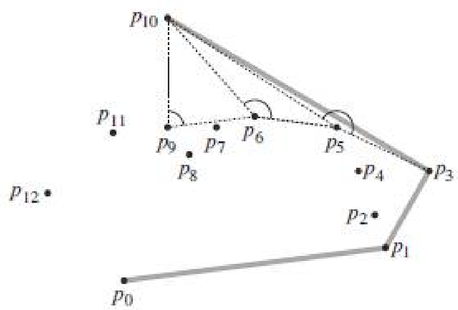

Fig.2.9[1]

In fig 2.8, p9 is pushed on the stack, here it makes non-left turn from p8 to p9. So p8 is popped out from the stack. In fig 2.9, p10 is pushed on to the stack, here it makes non-left turn from p9 to p10, p7 to p10, and p6 to p10 , so we pop p9,p7,p6 from the stack. It makes left turn from p5 to p10, so p5 should be there on stack.

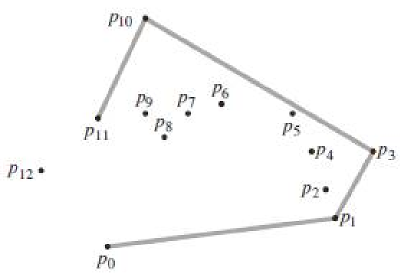

Fig.2.10[1]

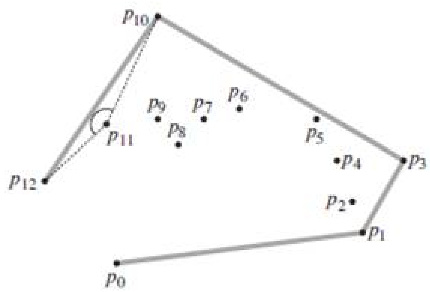

Fig.2.11[1]

In fig 2.10, p11 is pushed on to the stack, its not making non-left turn from the p10. So p11 should be on stack. In fig 2.11, p12 is pushed on to the stack , but it makes right turn , means non-left turn from p11. So p11 should not be there on stack.

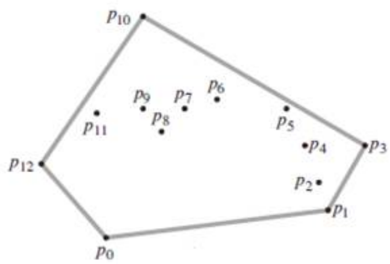

Fig.2.12[1]

The fig 2.1 to 2.12[1] shows the execution of Graham's scan algorithm. The fig 2.12 shows the final output of the algorithm. It shows the polygon which contains the points of set CH (Q).

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

B. *Jarvis's March algorithm:*

This algorithm is also called as gift wrapping algorithm [15]. In this algorithm we choose the point $p_{0,}$ which is on the left most side. It starts wrapping the outer most points of the set of input points Q. Finally it ends up by reaching the same point p_0.It chooses the points such that which makes the less polar angle in counterclockwise order. It means that it always choose the point which lies on the right most end. By traversing like this, it completes the right chain. In the case of left chain it chooses the polar angle which is minimum with respect to its negative x-axis. The following Fig 3 indicates the execution of Jarvis's March algorithm.

Jarvis's March (Q)
1 Let input points set Q and left most point be p_0.
2 Let p_1 be the point such that all other points lie to the right of p_0p_1.
3 i $\leftarrow$ 3
4 CH (Q) $\leftarrow$ $\{(\overline{p_0,p_1})\}$
5 repeat
MAX = 0
for all p∈Q do
6 if $\angle$(p0-p1-p2)>MAX then
MAX $\leftarrow$ if $\angle$(p0-p1-p2)
$p_0 \leftarrow p$
end if
7 end for
8 Append (p_i,p_0) to CH(Q).
9 until $p_i \neq p_1$

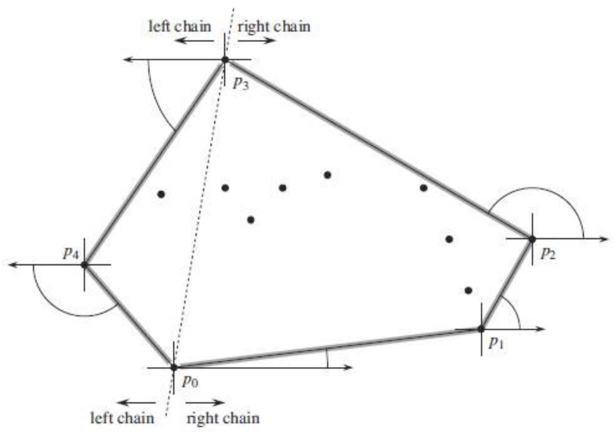

Fig 3. Execution of Jarvis's March algorithm

IV. ANALYSIS OF TIME COMPLEXITY OF ALGORITHMS

The time complexity [8][11][12][14] is the time taken to run the function of the algorithm. It is commonly expressed in O (Big o) notation. In this paper we are concentrating on the time complexity of Graham's scan[2] and Jarvis's March algorithms.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

A. *Time Complexity of Graham's algorithm:*

Here, let n be the number of input points. The Graham's scan algorithm takes O (nLogn) time. The 1st step in the algorithm, finding the point which is having less y co-ordinate takes O (n) time. The 2nd step, sorting points according to polar angles in increasing order takes O (nLogn) time. The next step stack operation takes O (n) time. To process the points one by one it takes O (n) time assuming that the stack operations takes O (n) time. So the overall complexity will be O(nLogn).This grahams scan algorithm is implemented and took various results by increasing the number of input points(n).The following Table.1 shows the number of input points and time taken to run the algorithm for various sets.

TABLE I
TIME COMPLEXITY OF GRAHAM'S ALGORITHM

n (num of input points)	**O(nLogn) Time in sec**
10	0.03
11	0.04
18	0.05
22	0.05
23	0.06

Fig 4 shows the time complexity graph. That is O(nLogn) curve. X-axis is the number of input points(n) and Y-axis is the time taken in sec to run the algorithm.

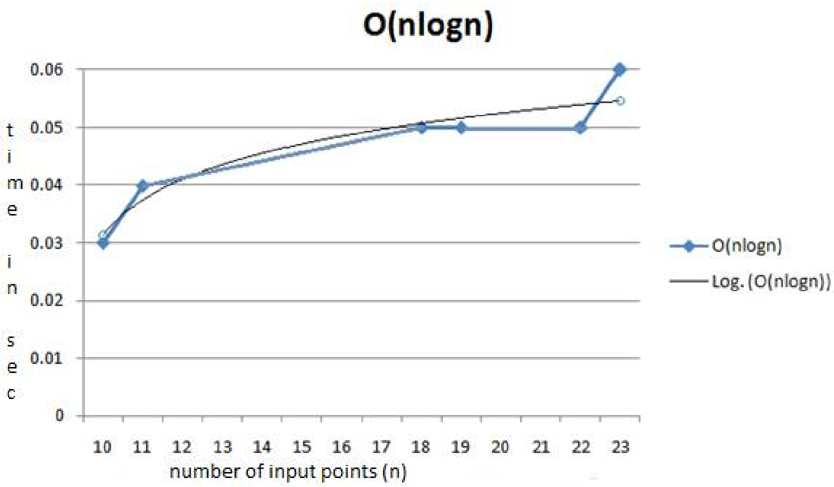

Fig.4 Time complexity graph of Graham's scan algorithm

B. *Time Complexity of Jarvis's March algorithm*

The Jarvis's March algorithm is having O (nm) time complexity. Where the n is number of input points and m is the number of points which forms the convex hull. In the worst case the time complexity will be O(n^2).Which means the number of input points are equal to the number of points forming the convex hull.

Jarvis's March algorithm is implemented and took results for various sets of input points. In Table II the results are tabulated.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

TABLE II
TIME COMPLEXITY OF JARVIS'S MARCH ALGORITHM

n (num of input points)	**m (num of output points)**	**O(nm) Time in sec**
7	5	0.02
8	6	0.02
9	6	0.02
10	6	0.03
11	6	0.03
12	6	0.04

Fig.5 shows the time complexity graph of Jarvis's March algorithm. The curve in the graph represents the O (nm) time complexity.

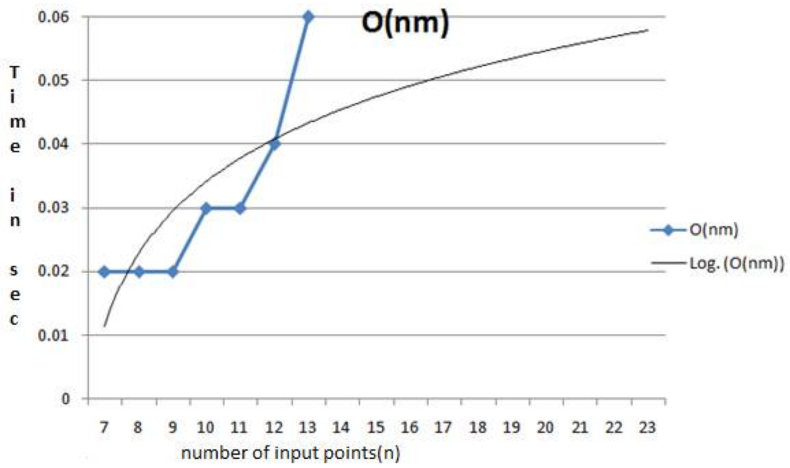

Fig.5 Time complexity graph of Jarvis's March algorithm

Here, the graham's scan algorithm takes 0.04 sec for 11 input points and Jarvis's March algorithm executes in 0.03sec for same number of input points. By comparing the results of two algorithms we can say that Jarvis's March algorithm is faster than Graham's scan algorithm.

V. **APPLICATIONS**

There are many application [6][7][10][13] of convex hull in various fields. Some are mentioned below-

A) *Image processing*

The convex hull algorithms are used to find the boundaries of objects in the images and also in segmentation of images.[6]

B) *Robotics*

These algorithms are used in robot motion planning to find the shortest path.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

C) *Nuclear/Chemical leak evacuation*

If there is any leakage in the Nuclear/Chemical industries, the sensors are placed to identify the leak. By using convex hull, we can find the shortest path to evacuate the place.

D) *Tracking the disease epidemic*

We can use the convex hull to find out the areas infected by the diseases.

E) *Image editing software*

For example Photoshop. In this software convex hulls are used to change the background of the image by separating the foreground image.

F) *To identify the area of rain fall*

Convex hulls are formed by taking the results of rain fall in various places to find out the total area covered by rain fall.

VI **APPLICATION OF CONVEX HULL IN IMAGE PROCESSING**

In image processing[5], the convex hull image can generate from the binary image. The convex hull helps to identify the foreground and background image.

In this paper, CT lung cancer gray scale image is used as input and applied various methods to generate convex hull image. The fig.6.0 is the original image of CT lung cancer. It is read and displayed in MATLAB.

Fig.6.0 Original Image CT lung cancer

The gray scale image is converted in to binary image by using MATLAB, before applying convex hull function. Fig6.1 shows the binary image.

Fig.6.1 Binary image of CT lung cancer

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

For binary image (fig.6.1) bwconvhull (BW) function is applied in MATLAB to get the image showed in fig.6.2. The fig.6.2 shows the union convex hull.

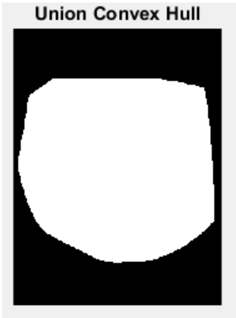

Fig.6.2 Union Convex Hull

The bwconvhull (BW,'Objects') is a function applied on binary image to get convex hulls of different parts of an organ present in the image. The fig.6.3 shows the objects convex hull image.

Fig.6.3 Objects Convex Hull

By applying convex hull on CT images, the boundaries of organs are identified in MATLAB. The segmentation of tumors from images is done.

VI. CONCLUSION

The Convex Hull problem is the widely used geometric problem in computational geometry. In this paper, the comparison is done for the time complexity of Graham's Scan and Jarvis's March algorithms. Found that Jarvis's March algorithm[1] works faster than Graham's Scan algorithm. The convex hulls are having many application in the fields like Image processing, Robotics and many more. An example of image processing application is shown in this paper by using MATLAB.

REFERENCES

[1] Thomas H Cormen,Charles E leiserson, Ronald L.Rivest, Clifforrd Stein,"Introduction to Algorithms", 3rd edition, The MIT PressCambridge, Massachusetts London, England,2009.

[2] Veljko Petrović, Dragan Ivetić, "Linearization of Graham's Scan Algorithm Complexity", ICIT , The 5th International Conference on Information Technology,2011.

[3] Jingfan Fan, Jian Yang_, Yitian Zhao, Danni Ai, Yonghuai Liu, Ge Wang, and Yongtian Wang, "Convex Hull Aided Registration Method (CHARM)", IEEE Transactions on Visualization and Computer Graphics,2016.

[4] Convex hull function in MATLAB(Online), Available: https://in.mathworks.com/help/images/ref/bwconvhull.html

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 6, Issue 4, April 2018

[5] Rafel C. Gonzalez,Richard E.Woods, "Digital Image Processing" , 3rd edition, Pearson publication,India,2009.
[6] Junyi Xu, Zuoya Zheng, Yanping Feng, Xihong Qing, "A Concave Hull algorithm for scattered data and its applications",IEEE 3rd International Congress on Image and Signal Processing,2010.
[7] GAO Yang, Yuhu Cheng;,Xuesong Wang, "A Quick Convex Hull Building Algorithm Based on Grid and Binary Tree" , Chinese Journal of Electronics,2015.
[8] Artem Potebnia,Sergiy Pogorilyy, "Innovative GPU accelerated algorithm for fast minimum convex hulls computation", IEEE Federated Conference on Computer Science and Information Systems (FedCSIS),2015.
[9] Xujun Zhou, Xianxia Zhang, Bingfei Zhang, "An Incremental Convex Hull Algorithm Based Online Support Vector Regression", IEEE Proceedings of the 34th Chinese Control Conference, Hangzhou, China,2015.
[10] Bo Shen Woun, Guat Yew Tan, Miin Huey Ang, Ya Ping Wong, "A Hybrid Convex Hull Algorithm for Fingertips Detection",11th International Conference on Computer Graphics, Imaging and Visualization,2014.
[11] Li Ruqiong Zhang Jiahuan Shen Huai Li Zhi, "An improved algorithm and implementation for three-dimensional convex hull", IEEE Fifth Conference on Measuring Technology and Mechatronics Automation,2013.
[12] LinnaHuang, Guangzhong Liu,"Proved Quick Convex Hull Algorithm for Scattered Points", IEEE International Conference on Computer Science and Information Processing (CSIP),2012.
[13] Azizul Azhar Ramli, Witold Pedrycz , Junzo Watada and Nureize Arbaiy, "A Real-Time Analysis of Granular Information:Some Initial Thoughts on a Convex Hull-basedFuzzy Regression Approach",IEEE International Conference on Fuzzy Systems June 27-30, 2011, Taipei, Taiwan,2011.
[14] Dong Wei, XingHua Liu, "Improved algorithm for computing Convex Hull of plane point set",IEEE,2009.
[15] L.Cinque, C.Di Maggio, "A BSP Realisation of Jarvis's Algorithm",IEEE 1999.