International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

Geometrical Analysis Of Snow Fall Region In Aerial View Using Image Scaling Factor & Geometrical Focal Length

M.Mayilvaganan#1, Associate Professor, Department of Computer Science, P.S.G College of Arts and Science, Coimbatore, India , 1dev7aki@gmail.com

R.Manickam*2 , Research Scholar(Ph.D), Department of Computer Science, P.S.G College of Arts and Science, Coimbatore, India

Abstract

In this study, the estimation of overall snowfall occurring region resulting from flight runway area with (heavy), (medium) and (light) snowfall as a background situations.
The output from the proposed methods implemented in this work predicted the estimation of overall snow fall coverage area and density of the snow fall occurrence and identify the best geographic location to land the flight. Several assessment methods investigated such as binarization of the image, RGB to Gray Conversion and removal of noise using median filtering. The proposed methodology reveals the splitting of snow fall region using density based segmentation, and clustering the pixels using density grid based clustering with scaling factors. The results of the above estimates are presented and compared to efficiency of the proposed methodology.
Final objective is to identify the suitable cluster density for landing the flight with respect to minimum density region and size for different sets of images in the snowfall occurrence area. The efficiency of the proposed methodology have been compared with DBSCAN and Data Grid algorithm with respect to memory usage and running time of the algorithms for different sets of aerial images.

Keywords — *Binarization, Clustering, Data Grid, Density, DBScan, Image Scaling factors.*

Introduction

The proposed methodology begins with the Image Acquisition and image pre-processing techniques. The image acquisition can be used to load the aerial images can be gathered from various flight runway area with snowfall as background situations. he images are stored in RGB color model, each colour appears in its major spectral components of red,green,and blue. The colour of a pixel is made up of three components; red, green, and blue(RGB), described by the corresponding intensities. In the RGB colour model, a colour image can be represented by the intensity function.

$$I\ RGB = (AR, AG, AB) \qquad (1)$$

Where AR(x,y) is the intensity of the pixel (x,y) in the red channel, AG(x,y) is the intensitty of pixel (x,y) in the greenchannel, and AB(x,y) is the intensity of pixel (x,y) in the blue channel. The intensity of each colour channel is usually stored using eight bits,which indicates that the quantization level is 256.
The Image Processing involves an assortment of steps namely; Image preprocessing, Restoration, Analysis and Compression. Pre-processing includes numerical correlation and radiometric correlation. The associated image is then fed for re-establishment task.
In this research work, an aerial digital images can be gathered from various flight run way occurrence with snowfall as background situations.
The organization of the paper is structured as follows. Chapter 2 explains about the research problem and Data for Research .Chapter 3 demonstrates the proposed methodology for DBSCAN clustering. Chapter 4 reveals the Results and Discussion for density clustering techniques. Finally, Chapter 5 concludes the paper.

About the Research Problem

From a risk organization position, snow on the runway is the foremost cause of snow-linked aircraft mishaps. Most mishaps occur throughout the landing phase, but snow is also known to bite pilots throughout takeoff and taxi operations. Earlier than taking the runway, it is forever best to remain for snow removal to occur, but not all little unattended airports have that ability.

Data For Research

In this research work, an aerial digital images can be gathered from various background situations with 7.2MP resolution. The original images where resized to a lower resolution of approximately 457x630 pixels so the algorithms chosen can process them more efficiently.
Figure 1 (a) and (b) shows the image datasets used for this study.

ISSN:2319-7900

Fig 1(a) set1 image with snow

Fig 1(b) set2 image with snow

Proposed methodology -Need for DBSCAN Clustering

In order to identify the best density location to land the flight, there is a need to segment the region using black and white pixels. Snowfall appearance region are considered as black pixels(1's). Area other than snow are considered as whitepixels(0's). The problem of segmenting the snowfall region pixels is difficult when the clusters are of dissimilar size, density and shape. Many of these issues become even more considerable when the data is of very high dimensionality and when it includes noise and outliers.

DBSCAN Clustering Implementation

To clusters a dataset, our DBSCAN implementation starts by identifying the k nearest neighbours of each point and identify the farthest k nearest neighbour (in terms of Euclidean distance)1. The average of all this distance is then calculated. After that, for each point of the dataset the algorithm identifies the *directly density-reachable* points (using the Eps threshold provided by the user) and classifies the points into *core* or *border* points.

Pseudocode steps for DBSCAN clustering

1) Start with an arbitrary starting point that has not been visited.
2) Extract the neighborhood of this point using ε (All points which are within the ε distance are neighborhood).
3) If there are sufficient neighborhood around this point then clustering process starts and point is marked as visited else this point is labeled as noise (Later this point can become the part of the cluster).
4) If a point is found to be a part of the cluster then its ε neighborhood is also the part of the cluster and the above procedure from step 2 is repeated for all ε neighborhood points. This is repeated until all points in the cluster is determined.
5) A new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.
6) This process continues until all points are marked as visited.

DBscan Clustering Flowchart

The following diagram(2) represents the flowchart for density based clustering for grouping of snowfall region according to black and white pixels in the flight runway area.

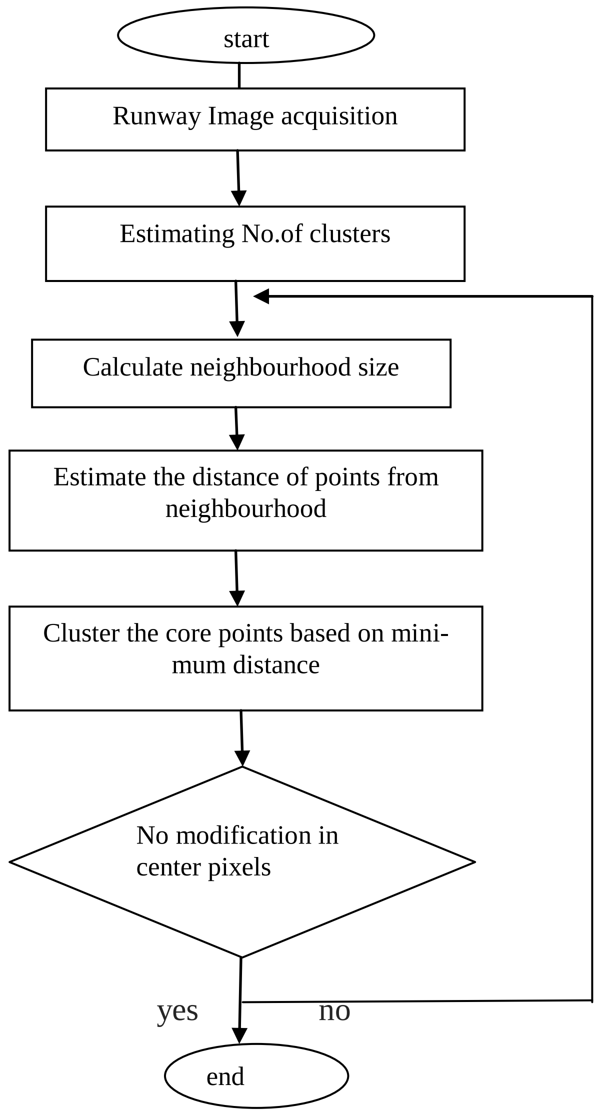

fig 2 DBscan clustering flowchart

The main procedures of the module that implements the DBSCAN algorithm are:

ISSN:2319-7900

1.CheckDirect Density Reachable Points() that identify the *total no. of black pixel* points,
2. FindClusters() that start to classify the points into clusters with the support of the GetDensity Reachable Points() procedure that identify the *black pixels with similar size.*
3. VerifyClusters() procedure that verify the composition of the nearest best density clusters to land the flight safely in snowfall occurrence area..

DBSCAN Mathematical Model

Consider a various aerial digital images (n) can be gathered from flight run way occurrence with snowfall as background situations. To clusters a dataset, our DBSCAN implementation starts by identifying the k nearest neighbours of each point and identify the farthest k nearest neighbour (in terms of Euclidean distance ki).

Method for finding value of Euclidean distance 'ki'

To find the value of ki automatically, consider a image dataset with n points. First we will have to find out all the points average one to all other points distance to other points.
Let's consider O1 and O2 be two objects from the aerial digital images from the universe of possible objects.
The distance between O1 and O2 is denoted by distance (O1,O2) or d(O1,O2).
The joining or tree clustering method uses the disimimalrities or distance between objects when forming clusters. It can be represented using Euclidean distance as follows.

$$\textbf{Euclideandistance}(Oi,Oj)=2\mathrm{sqrt}(\Sigma^n(Oik-Ojk)^2 \qquad (2)$$

$= \mathrm{sqrt}(5-3)2 + (6-9)2 + (4-3)2 + (9-2)2$

$=8.25$ cm

The average of all this distance is then calculated. After that, for each point of the dataset the algorithm identifies the *directly density-reachable* points (using the Eps threshold provided by the user) and classifies the points into *core* or *border* points.

Method for finding value of Average distance 'ai'

This distance is simply the average difference across dimensions. It can be represented using the following formula.

$$\textbf{Distance}(Oi,Oj) = 1 / n\ \Sigma^n\ |\ O_{ik} - O_{jk}\ | \qquad (3)$$

$= I / 4 (|5 - 8| +|6-9|+|4-3|+|9-2|$

$= 3.5$ cm

Method for finding ε -Eps value

Let x and y be objects in Fd, a d – dimensional input space for the image dataset and F be the influence function of data object y on x is a function,

$F_b{}^y : f^d \rightarrow R_o+$, which defines in the terms of a basic influence function f_b

$$F_b{}^y(x) = f_b(x,y) \qquad (4)$$

In principle, the influence function can be an arbitrary function that can be determined by the distance between two objects in a neighbourhood.
Euclidean distance can be used to compute a square wave influence function, it can be represented as follows.

$F_{square}(x,y) = \{ o$ if $d(x,y) > \sigma \quad \{ 1$ otherwise

Or a Gaussian influence function,

$$F_{gauss}(x,y) = e - d(x,y)^2 / 2\sigma^2 \qquad (5)$$

The density function at an object or point x ε Fd, the density function at x can be defined as

$$F_b{}^d(x) = \Sigma_n{}^{i=1} f_b{}^{xi}(x) = f_b{}^{x1}(x) + f_b{}^{x2}(x)...+f_b{}^{xn}(x)$$

Method for determining Minpts values

After determining the different Eps values, there is a need to estimate the value of the MinPts is the immediate and urgent task. So firstly, the number of data objects in Eps neighborhood of every point in dataset is calculated one by one. And then mathematic expectation of all these data objects is calculated, which is the value of MinPts.

$$\textbf{Minpts} = 1/n\ \Sigma^n{}_{i=1} P_i \qquad (6)$$

Where pi is the number of points in Eps neighborhood of point i. So for each different value of Eps we will get corresponding Minpts value.

Method for finding value of 'k'

To find the value of k automatically, consider a dataset with n points. First we will have to find out all the points average one to all other points distance to other points. Let's consider one point and find distance to all the other points from it and average it to find the average distance.

$$D\ (_{Pi}) = \Sigma^n{}_{i=1}\ \textbf{distance}\ (P_i\ 2_i) / 2(n-1) \qquad (7)$$

Here, d(Pi)=Average distance from Pi to all other points in the data set.
And determine the d(Pi) for all Pi and avg(d). Which is the average of all d(Pi) which is required to find out the Target Point (Ti) .,

$$\textbf{Avg}(d) = \Sigma^n{}_{i=1}\ d\ (Pi\ ) / n \qquad (8)$$

For every Pi in the datasets we will draw a circle and the centre of the circle will be the points itself means Pi, and the radius of each circle will be the avg(d). So area of each circle will be same. Here we conceive only the circumference of each circle. (3)
Here,
Pi=Subjective Point or Centre of the Circle
r =avg(d) (Radius of each Circles.)
For every circle we have to determine the closest point which is nearest to the circumference of each circle by the following equation

$$\min|\ (\textbf{distance}\ (\ r - xi\ ))\ | \qquad (9)$$

(4)

ISSN:2319-7900

Xi is the point which has minimum distance from the circumference of a particular circle for the corresponding Pi which is the centre of that circle.

Ti(Pos)=Position of the Ti relative to the Pi of a particular circle.

If there is more than one mode then there is a necessity to compute the mean of maximum repeated Ti(Pos)s or modes. Mode of Ti(Pos) is basically our expected value of parameter K in the K-dist plot .

IV. RESULTS AND DISCUSSION

DBSCAN then iteratively collects directly density-reachable objects from these core objects, which may involve the merge of a few density-reachable clusters. The process terminates when no new point can be added to any cluster.

The following diagrams(3) and (4) illustrates the Original image of snowfall runway area before clustering and after clustering with eps and minpts value. After clustering the snowfall area, the image scaling factors are estimated. In order to determine the density of the snowfall area, eps and minpts values are been estimated using DBSCAN algorithm.

Fig 3 original aerial image

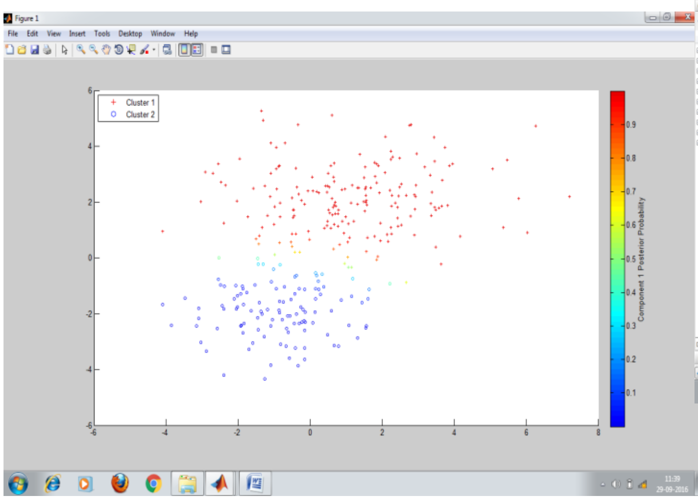

Fig 4 DBSCAN clustered image

Using the binarization and scaling factors of the image, snowfall appeared region are estimated as black pixels(1's) and other area are considered as white pixels(1's). The following figure represents the estimation of black pixels 41,236 and snowfall density clustering height 0.0024250 mts and total snowfall coverage area 0.989688 hectares respectively.

After segmentation and clustering of image, then the image is ready for the estimation process such as image scaling and total are calculation. The following fig (5) and (6) represents the snowfall density clustering height, total no.of black pixels and total area coverage of snowfall occurrence in flight flyover area. Hence the overall snowfall occurrence area can be calculated using the following formula,Total occurrence of snowfall Real area = primary area * m.

Image scaling(I) = focal length(fl) of the camera / height(h) of snowfall clustering.

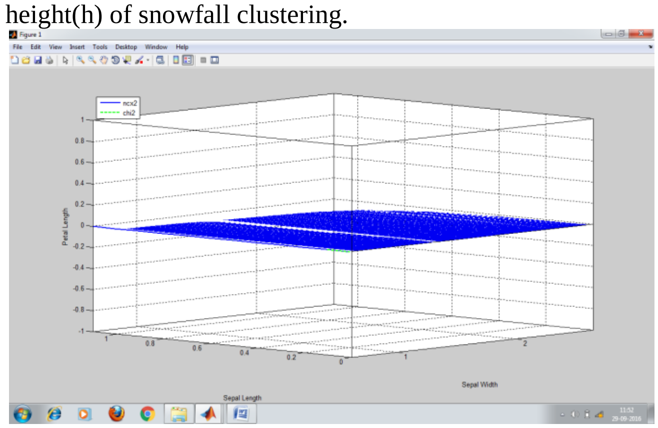

Fig (5) DBSCAN estimation of eps and minpts for aerial image dataset1

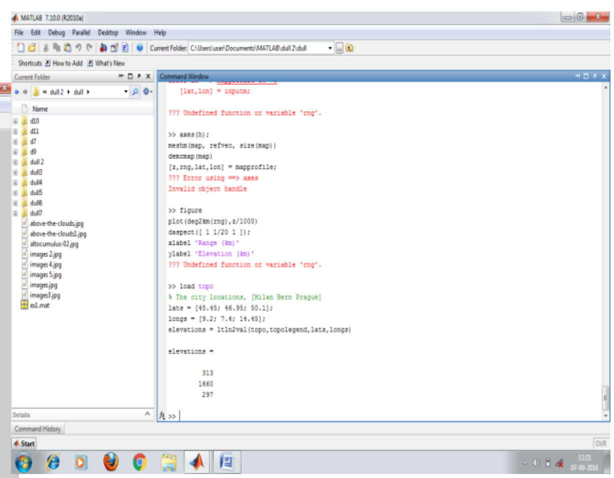

Fig 6 DBSCAN estimation of snowfall density and primary area coverage

V CONCLUSION & FUTURE WORK

This chapter briefly describes the proposed methodology which is useful for analyzing and estimating the cluster segmentation of snowfall area using DBSCAN algorithm. The performance of algorithm was analyzed from the estimating

ISSN:2319-7900

the black pixels in snowfall area with heavy snow/ medium snow and ordinary snow using DBSCAN algorithm. The true image binarization and black pixels Count are estimated using binary conversion technique. The proposed design can be used for analyzing the density of snowfall occurrence and estimated the spread over circumference area of snowfall region , and a framework of methodology has been developed for analyzing the aerial image sequence for a step by step process. Regarding the possibilities of future research on the same lines, the current research can be extended to and evaluate the work in the following areas:

1. Extend the system to estimate the density of snowfall region and estimate the total coverage of snowfall region.
2. Extend the proposed methodology focal length of the camera above 100mm.

This promises a great scope for further research on these lines.

REFERENCES

[1]. Density-Based Clustering Algorithms, research work, Lahore College for Women University Lahore, Pakistan.

[2]. Pasi Fränti. Number of clusters (validation of clustering), Speech and Image Processing Unit School of Computing University of Eastern Finland.

[3] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. SIGMOD'98.

[4]. C. Rother, V. Kolmogorov, T. Minka, and A. Blake. Cosegmentation of image pairs by histogram matching – incorporating a global constraint into mrfs. In *CVPR*, 2006.

[5]. Russell, W. T. Freeman, A. A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[6]. J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pat-tern Analysis*. Camb. U. Press, 2004.

[7]. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. PAMI*, 22(8):888–905, 1997.

[8]. J. Winn and N. Jojic. Locus: learning object classes with unsupervised segmentation. In *ICCV*, 2005.

[9]. L. Xu, J. Neufeld, B. Larson, and D. Schuurmans. Maximum margin clustering. In *NIPS*, 2005.

[10].K. Mumtaz1 and Dr. K. Duraiswamy. A Novel Density based improved k-means Clustering Algorithm – Dbkmeans. (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 02, pp. 213-218, 2010.

[11]. Duchenne, I. Laptev, J. Sivic, F. Bach, and J. Ponce. Automatic annotation of human actions in video. In *ICCV*, 2009.

[12]. P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graphbased image segmentation. *International Journal of Computer Vision*, 59, 2004.

[13].M. X. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.

[14].T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2001.