MAUSAM, **66**, 4 (October 2015), 695-712

551.501.815 : 551.578.7

Radar imageries information extraction and its use in pre-hail estimation algorithm

P. KUMAR and DEBA PRASAD PATI

National Initiative on Climate Resilient Agriculture Project, MIT College of Engg., Pune – 411 029, India

(*Received 25 March 2014, Modified 17 September 2014*)

e mail : pkumarmbm@yahoo.com

सार – एक बहुत बड़े डॉप्लर मौसम रेडार (DWR) चित्र संसाधित उपकरण विकसित किया गया और PHDA में इसकी उपयोगिता का परीक्षण कर इसे प्रामाणिक माना गया। ओला नियंत्रण प्रक्रिया के लिए परावर्तकता (ART परिकलन के लिए) समय और समन्वय (दिशा और गति परकलन के लिए) वाले रेडार चित्रों के ROL से सूचना प्राप्त करने की आवश्यकता होती है जिससे PHDA को सूचनायें मिलती हैं। इस शोधपत्र का मुख्य उद्देश्य इन सूचनाओं को प्राप्त कर उचित आँकड़ा बेस में संग्रहित करना है ताकि PHDA द्वारा उन्हें कुशलतापूर्वक संसाधित किया जा सके। इन सूचनाओं को 'एक्सेल फाइल' में संग्रहित किया जाता है ताकि परिणामों को प्रस्तुत करने के लिए PHDA द्वारा किसी भी प्रकार की अपेक्षित सूचनाओं को पुन: प्राप्त कर तत्काल उनका विश्लेषण किया जा सके।

जिस समय ओला मेघ में वृद्धि हुई उस समय नागपुर DWR आँकड़ों से प्राप्त दो मामलों का विश्लेषण किया गया। इससे प्राप्त सूचना को PHDA द्वारा संसाधित किया गया और प्राप्त परिणामों की तुलना की गई तथा वास्तविक ओला की घटना के आँकड़ों सहित इस पर विचार विमर्श किया गया।

ABSTRACT. A comprehensive Doppler Weather Radar (DWR) image processing tool has been developed and its use in Pre Hail Detection Algorithm (PHDA) is tested and validated. For hail control operation, the information needed to be extracted from Region of Interest (ROI) and Point of Interest (POI) of radar imageries are reflectivity (for Available Reaction Time 'ART' calculation), time and co-ordinates (for direction and speed calculation) which are the inputs to PHDA. This paper focuses mainly to extract these information and store them in proper data base so that they can be efficiently processed by PHDA. The information is stored in an 'Excel file' so that any required information can be retrieved and analyzed immediately by PHDA to produce the result. The entire process provides an automatic environment for PHDA operation and working.

Two cases from Nagpur DWR data, when the growth of hail cloud was noted, are analyzed and the information obtained are processed by PHDA and the result is compared and discussed with actual hail occurrence data.

Key words – Pre hail estimation algorithm, Hailstorm, Region of interest, Point of interest, Image processing.

1. Introduction

The process of radar imaging is similar to traditional radar, in that it involves sending out electromagnetic pulses. Backscatter from the pulses is examined more thoroughly and is subjected to a process referred to as radar image processing. There are a few different ways to examine radar backscatter in order to extrapolate an image, though they typically involve calculating the scattering coefficient of the areas that are under observation. Radar image processing breaks up these areas into pixels, or picture elements, and the precise behavior of the backscatter is used to extrapolate an image (Mandhare *et al*., 2013).

Advent of weather radars during the World War-II proved boon to researchers in the field. However, most of the weather radar related software addressed to the prediction of hailstorm, rather than prediction of time taken by the cumulus cloud to turn into hailstorm (Kumar, 2010). This information is important as during this interval only if CCN's are released in to the cloud then only we can restrict the larger growth of hail kernels. This period is known as "Reaction Time". Total Reaction Time (TRT) may be defined as the time taken by any cumulus cloud with reflectivity 20 dBZ to grow till its reflectivity reaches 45 dBZ. Available Reaction Time (ART) is the time actually available within the TRT for action against the threatening cumulus cloud growth. This paper described the methodology to extract important information from radar imageries, their proper storing and automatic access by Pre Hail Detection Algorithm (PHDA), developed by authors.

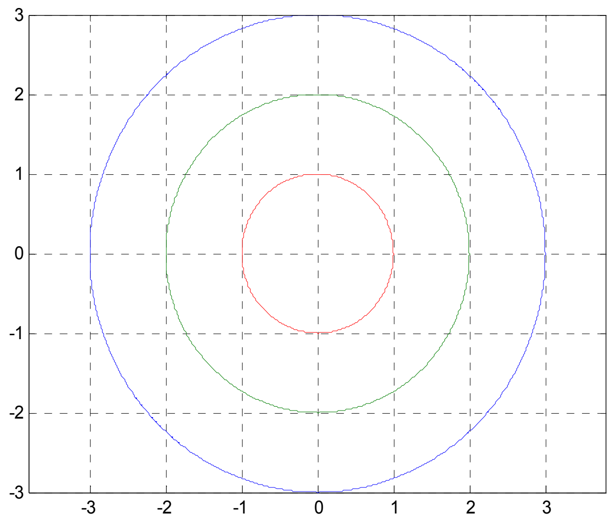


Fig. 1. MATLAB output for concentric circles

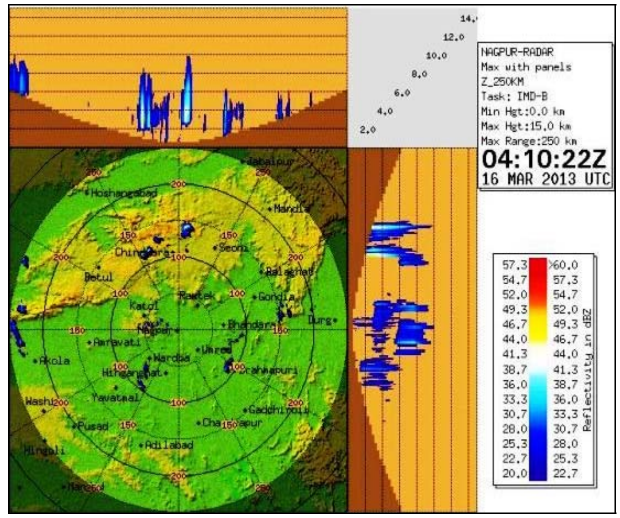


Fig. 2. Radar image to be overlapped and analyzed

2. Data

From Nagpur and Patna, India DWR imageries of radar reflectivity was collected at 10 minutes interval for 10 elevations from 0.2° to 21° on 16th March, 2012 and 1st May, 2012. DWR was M/S Beijing Metstar made single polarization mode, S-Band (≈2800 MHz) with PRF of 200-1200 Hz (selectable). It was operated at 2 RPM with volume scan repeated in 10 minutes. These imageries are analyzed to extract the information like Reflectivity, location and time.

3. Methodology

The images used for analysis in this paper are mainly "IRIS software" based images which contains information like reflectivity, range, azimuth etc. Information such as co-ordinates, reflectivity, and time are extracted from the images by mouse clicking action on the image in a basic software environment. This data can be analyzed to determine the functionality and feasibility of PHDA. Data extraction and analysis from radar image includes following steps:

- Design and development of basic software environment to load the radar image for processing.
- Radar image pixel-by-pixel processing to obtain reflectivity values from radar images.
- Information extraction of any point of interest on the radar image by mouse clicking action.
- Storing of information in suitable file for analysis.
- Linking the data to the developed algorithm for optimization.

3.1. *Design and development of basic software environment for radar image for processing*

Clouds can be identified by means of RADAR reflectivity values which are the processed output of IRIS software installed with Weather Doppler Radar processor (Blaschke, 2010). In this paper the images used for analysis are already system generated images, hence a suitable software environment is needed to load the images and to extract information.

Fig. 1 shows the basic design of concentric circles to overlap with the radar image as shown in Fig. 2 on it, for processing so that the co-ordinates can be extracted.

3.2. *Radar image pixel-by-pixel processing to obtain reflectivity values from radar images*

To obtain the reflectivity information from a radar image, its pixel by pixel analysis for Red, Green and Blue (RGB) content is needed.

In digital imaging, a pixel or picture element is a physical point in a raster image, or the smallest addressable element in an all points addressable display device; so it is the smallest controllable element of a picture represented on the screen. The address of a pixel corresponds to its physical coordinates. LCD pixels are manufactured in a two-dimensional grid, and are often represented using dots or squares, but CRT pixels correspond to their timing mechanisms and sweep rates.

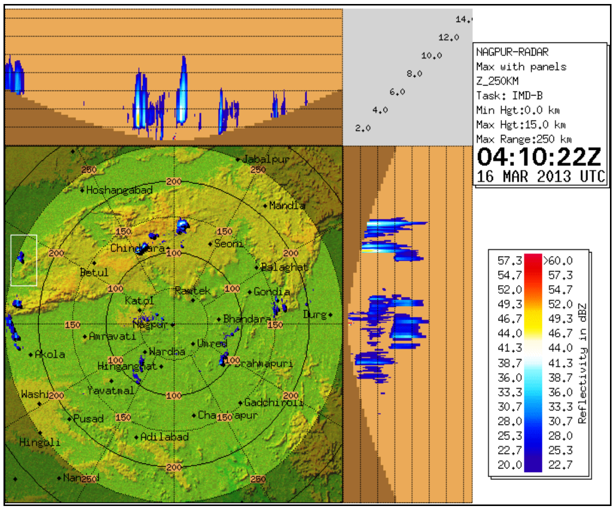


Fig. 3. Cluster inside the rectangular shed with Actual Reflectivity:21 dBZ

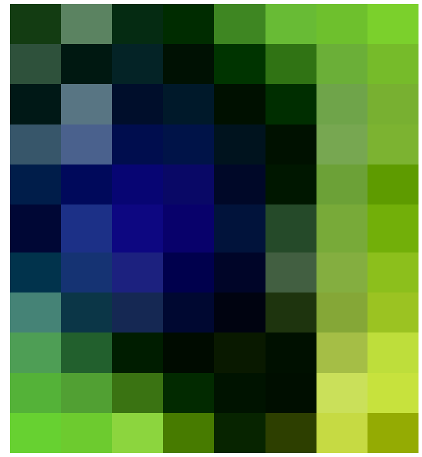


Fig. 4. Cropped ROI of Fig. 3 with increased reflectivity

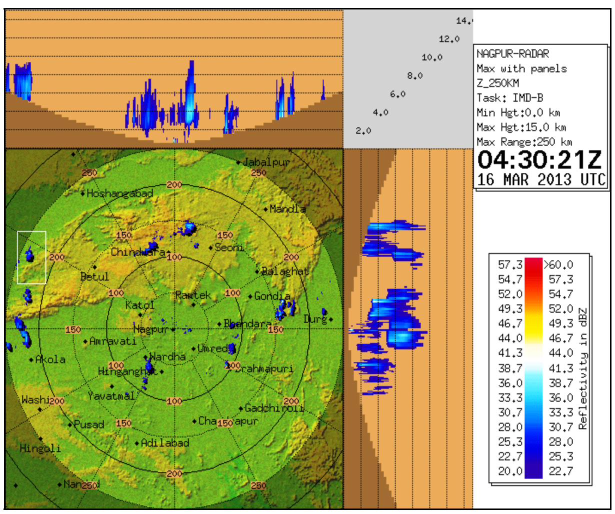


Fig. 5. Cluster inside the rectangular shed with Actual Reflectivity:27 dBZ

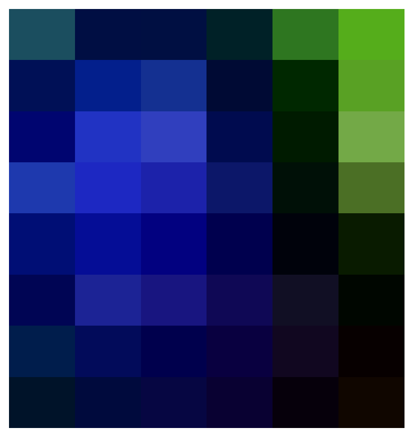


Fig. 6. Cropped ROI of Fig. 5 with increased reflectivity

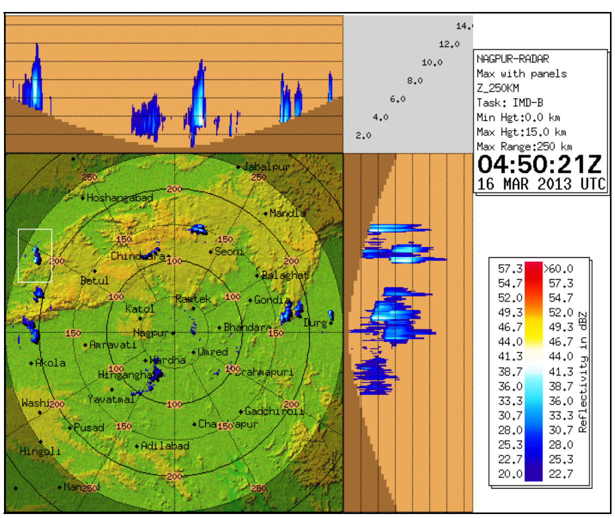


Fig. 7. Cluster inside the rectangular shed with Actual Reflectivity: 35dBZ

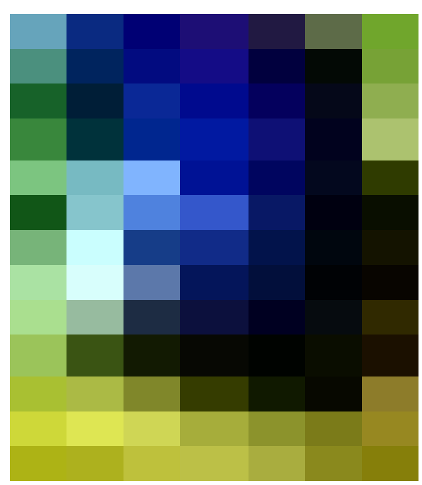


Fig. 8. Cropped ROI of Fig. 7 with increased reflectivity

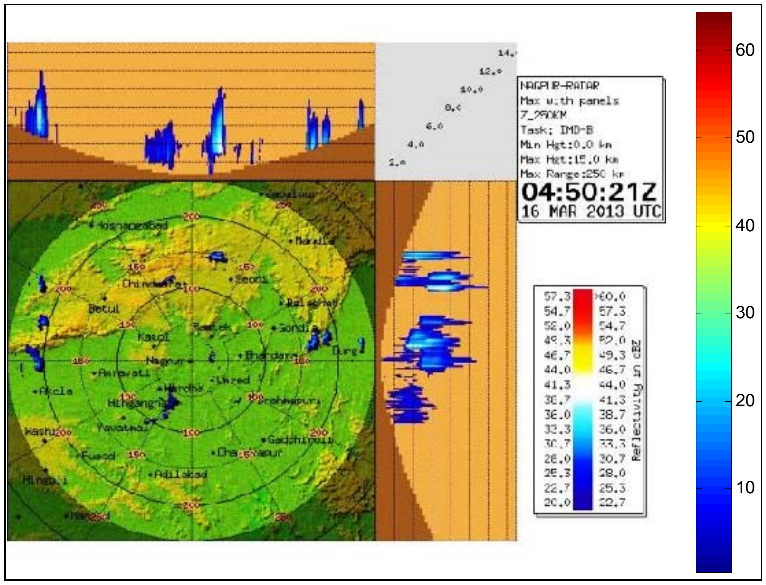


Fig. 9. Radar image with designed colour-bar

Each pixel is a sample of an original image; more samples typically provide more accurate representations of the original. The intensity of each pixel is variable. In colour image systems, a colour is typically represented by three or four component intensities such as red, green, and blue, or cyan, magenta, yellow, and black as per MATLAB Graphics.

Here in this paper, pixel analyses of coloured radar images are done mainly to get information about red, green and blue content.

A special algorithm is developed in "MATLAB" to get the colour value of particular Region of Interest (ROI) from the images (Gonzalez *et al.,* 2012). For clarity flow chart is shown in Appendix-A and Source program is shown in Appendix-B.

3.2.1. *Algorithm for reflectivity extraction from radar imageries*

The algorithm to extract the "reflectivity pattern" is developed by writing a program in MATLAB which is based on the logic of "ROI based pixel analysis" described as below (Gonzalez *et al*., 2012).

The radar image is composed of various colour patches with specific ranges where each range indicates a particular reflectivity.

In this paper three images of Nagpur based DWR are taken for analysis. The images are "ngp1.jpg","ngp2.jpg" and "ngp3.jpg".

After loading the image, ROI is cropped or selected for RGB value extraction.

For a cropped portion of the image the average of R (red), G (Green) and B (Blue) value is calculated and stored in some excel file.

Similarly for all three images average of RGB is calculated and stored in the excel file.

To analyze the ROI the ROI should be cropped by the help of MATLAB crop tools as described below:

"I = imcrop" creates an interactive Crop Image tool associated with the image displayed in the current figure, called the target image. The Crop Image tool is a moveable, resizable rectangle that you can position interactively using the mouse. When the Crop Image tool is active, the pointer changes to cross hairs '+' which can be moved over the target image. Using the mouse, "crop rectangle" can be specified by clicking and dragging the mouse. Crop rectangle can be moved or resized by the mouse. When sizing and positioning of the crop rectangle is finished, the cropped image can be created by double-clicking the left mouse button or by choosing "Crop Image" from the context menu (Gonzalez *et al*., 2012).

Figs. 3, 5 and 7 are the DWR Nagpur images used for analysis and Figs. 4, 6 and 8 are the ROI (Region of Interest) of the images as mentioned above.

From Fig. 4, Fig. 6 and Fig. 8, the average R, G and B values are computed and also the mean(RGB) is computed and stored. These values can be used for analysis and detail is described in the analysis part of this paper.

Similarly through "impixel()" command particular point on the ROI can be selected and its mean RGB value can be obtained (Detail is in the analysis part).

Though it is difficult to match the actual average RGB value with the colour range specified for reflectivity in the datasheet, so one new color bar is designed to implement the true average RGB value in terms of colour bar patches so that the coloured value obtained from the ROI point of interest would be processed further as true reflectivity.

Fig. 9 shows the radar image with the default color bar designed in MATLAB to extract the reflectivity in terms of specific value taken from default colour bar.

The skill score of predictions made by this method is 0.76. Accuracy of >0.75 is well within the reasonable acceptable limits of the prediction.

3.3. *Information extraction of any Point of Interest (POI) on the radar image by mouse clicking action*

PHDA (Pre Hail Detection Algorithm) mainly includes following sub algorithms:

- Detection of specific cloud based on the reflectivity value.
- Analyzing the growth of the same cloud.
- Finding out the "Available Reaction Time (ART)".
- Finding out the expected speed of the cluster.
- Finding out the expected direction or the heading of the cloud.

Hence the information is needed to be extracted from radar images are reflectivity value, time and co-ordinates from the Point of Interest (POI) within any ROI of image.

3.3.1. *Estimation of ART (Available Reaction Time)*

Available Reaction Time (ART) is the time actually available within the TRT (Total Reaction Time) for action against the threatening cumulus cloud growth. Hence to calculate ART we need the Reflectivity *vs* time values from the radar image. To facilitate the above process one program is written to display and store the time of mouse click on the specific Point of Interest (POI) on ROI. Mouse click is the time of spotting of specific clutter by the radar operator who feels that the cloud could grow into higher reflectivity. The assessment and intuition of the radar operator is based the daily updates on synoptic and climatological information on region and season and diurnal variation.

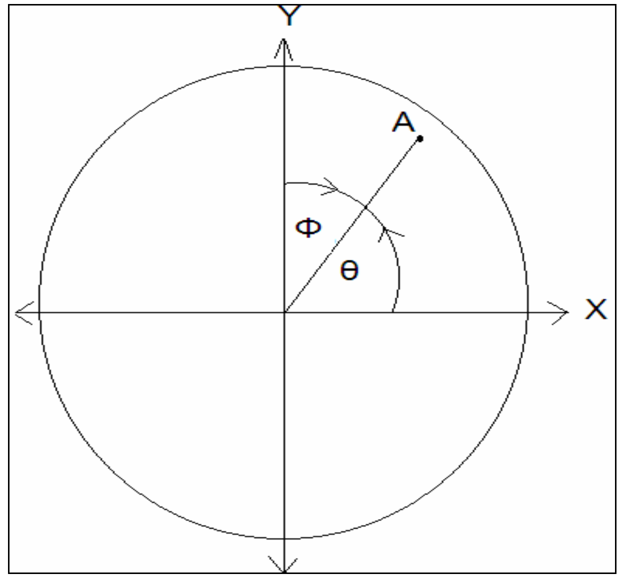


Fig. 10. Relation between θ and φ in first quadrant

3.3.2. *Speed and direction calculation*

Speed and direction of the threatening cloud need to be calculated as the information would help the pilot to chase the suspicious cloud for seeding.

3.3.2.1. *Computation of x, y and θ from r and φ*

3.3.2.1.1. *Conversion of φ^0 to θ radians.*

If φ is the azimuth angle in degrees, measured clockwise from north and θ the angle made by the target with the positive x axis (anticlockwise) then for first quadrant, as shown in Fig. 10 $\theta = \frac{\pi}{2} - \varphi_r$ and for 2nd, 3rd and 4th quadrant $\theta = \frac{5\pi}{2} - \varphi_r$; where $\varphi_r = \frac{\pi\varphi}{180}$ and φ_r is in radians and φ is in degrees.

3.3.2.1.2. *Conversion of r in x and y components.* $x = r \cos\theta$ *and* $y = r \sin\theta$

3.3.3. *Estimation of cloud speed*

The speed is computed at the midpoint of two time observations by dividing the linear distance between

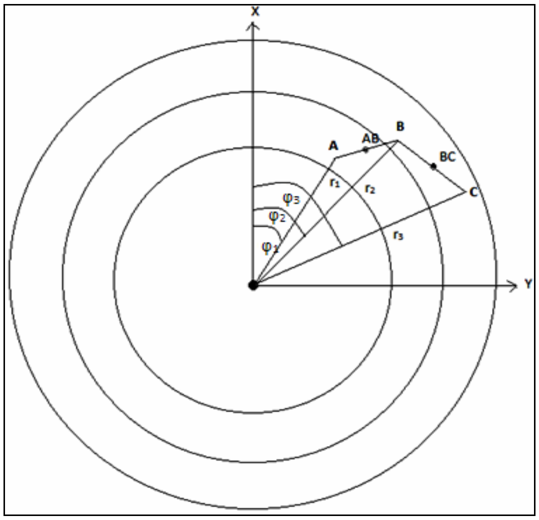


Fig. 11. Cloud cluster location on PPI display at different time

the two points by the time interval. The locations of points A (r_1, φ_1), B (r_2, φ_2), C (r_3, φ_3) are as shown in Fig. 11 and the time instances associated with point A (r_1, φ_1) is t_1 and the range and azimuth are r_1 and φ_1 respectively.

Similarly at point B (r_2, φ_2), time associated is t_2 and the range and azimuth are r_2 and φ_2 respectively and at point C(r_3, φ_3) time associated is t_3 and the range and azimuth are r_3 and φ_3 respectively.

Then at time $T_{AB} = \frac{t_1 + t_2}{2}$, speed at AB is $v_1 = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}}{t_2 - t_1}$ and speed at BC at time $T_{BC} = \frac{t_2 + t_3}{2}$ is $v_2 = \frac{\sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2}}{t_3 - t_2}$.

Hence, speed at time t_3 is given by

$$\left(\frac{v_2 - v_1}{T_{BC} - T_{AB}}\right) T_3 + C \text{, where, } C = \left(\frac{v_2\, T_{AB} - v_1\, T_{BC}}{T_{AB} - T_{BC}}\right).$$

Hence, as described above for both speed and direction computation we need mainly the x and y co-ordinates of the POI which can be converted to (r, φ) and *vice-versa*.

The process to get the information of co-ordinates by the mouse point click on POI is done by a special program in MATLAB.

3.4. *Storing of information in suitable file for analysis*

For further analysis of the extracted information, they should be stored in a proper file. So, all the information are stored in an "excel file" immediately after execution of the main program from which required information can be retrieved to execute the main program of PHDA.

3.5. *Linking the data to the developed algorithm for optimization*

Once all required information are stored in the excel file, the main PHDA program can be run to extract required information from the specific location of the excel file as inputs and execute the same to provide outputs.

4. Analysis

Based on the available radar data of 16th March, 2013 (Nagpur Radar), total six different images of same growing cluster were analyzed to validate the algorithm. Two different cases are analyzed.

4.1. *Case-1*

Here three different images of same growing cluster as shown in Figs. 4, 6 and 8 are used for analysis. The detailed analysis is given as below:

4.1.1. *Procedure*

Fig. 4 shows the cropped image which can be analyzed based on ROI (Region of Interest). ROI based analysis deals with processing of entire cropped region of interest through MATLAB image processing tools. ROI analysis is to get only the reflectivity of desired region which is the mean value of actual RGB content of all pixels associated with it (expressed as C).

POI based image processing is to get the required information such as position (x, y co-ordinates), time of mouse click on POI of ROI and reflectivity (Z1) of the specific point on ROI. After a single click on any point of ROI such above mentioned information are extracted and stored in the excel sheet from where these information can be automatically fetched by PHDA which is in-built in the main MATLAB program.

Tables 1-3 show the extraction of information from the specific radar image and Table 4 shows the excel sheet with required information being extracted and stored. Tables 5-7 show the analysis of a non-growing cloud.

After extraction of information, PHDA is used to find out the outputs and outputs are compared based on actual data.

4.1.2. *Extraction of reflectivity*

To extract and analyze reflectivity from radar imageries, two different cases are considered.

From Figs. 4, 6 and 8, the ROI and POI based "RGB" analysis are described as below:

Description of variable:

*I*2 (:,:, 1) = ROI matrix indicating Red component of all individual pixel associated with it.

*I*2 (:,:, 2) = ROI matrix indicating Green component of all individual pixel associated with it.

*I*2 (:,:, 3) = ROI matrix indicating Blue component of all individual pixel associated with it.

x = mean (*I*2 (:,:, 1) = Average Red value of all individual columns of pixels of ROI matrix.

y = mean (*I*2 (:,:, 2) = Average Red value of all individual columns of pixels of ROI matrix.

z = mean (*I*2 (:,:, 3) = Average Red value of all individual columns of pixels of ROI matrix.

w = mean (*x, y, z*) = Average Red, Green, Blue value of all individual columns of pixels of ROI matrix.

v = Average of *w* = Average RGB value of ROI.

C = *v* as a whole number.

P = pixel value of POI on ROI.

*Z*1 = Average value of P (POI) as a whole number on ROI.

TABLE 1

ROI analysis for Fig. 4

*I*2(:,:,1)-Red	*I*2(:,:,2)-Green	*I*2(:,:,3)-Blue	*v*	*C*	P for POI	*Z*1
88 0 0	117 14 25	131 43 42				
74 0 0	97 13 19	141 78 72				
0 7 9	9 5 8	91 115 102	44.1270	44	6 1 107	38
28 13 8	48 7 1	135 129 107				
21 28 0	51 33 0	115 127 77				
11 21 0	54 40 8	71 83 49				
34 0 0	96 29 11	45 0 0				

Values of different variables as described earlier are given as:

x = 36.5714 9.8571 2.4286

y = 67.4286 20.1429 10.2857

z = 104.1429 82.1429 64.1429

w = 69.3810 37.3810 25.6190

v = 44.1270

C = 44

P = 8 1 107

*Z*1 = 38

So, the average ROI and POI obtained after analysis are 44 and 38 respectively.

TABLE 2

ROI analysis for Fig. 6

*I*2(:,:,1)-Red	*I*2(:,:,2)-Green	*I*2(:,:,3)-Blue	*v*	*C*	P for POI	*Z*1
0 3 20	16 31 48	86 140 145				
0 33 48	5 51 63	112 195 190				
30 29 28	57 40 34	174 194 170				
0 5 2	14 13 1	117 150 128	53.0159	53	40 7 127	58
0 28 24	5 35 21	84 149 128				
0 2 0	29 11 0	76 90 77				
0 0 6	19 10 6	41 61 66				

Values of different variables as described earlier are given as:

x = 4.2857 14.2857 18.2857
y = 20.7143 27.2857 24.7143
z = 98.5714 139.8571 129.1429
w = 41.1905 60.4762 57.3810
v = 53.0159
C = 53
$Z1$ = 58

So the average ROI and POI obtained after analysis are 53 and 58 respectively.

TABLE 3

ROI analysis for Fig. 8

*I*2(:,:,1)-Red	*I*2(:,:,2) Green	*I*2(:,:,3)-Blue	*v*	*C*	P for POI	*Z*1
111 96 22	190 169 82	159 162 90				
135 93 0	200 144 30	228 199 83				
127 58 0	182 90 0	247 191 85				
73 29 14	128 55 9	192 166 125				
102 10 0	164 42 0	187 129 116	96.6667	96	93 144 199	145
75 0 2	144 36 11	126 94 128				
23 0 10	98 30 40	41 55 150				
57 0 0	135 50 38	60 59 143				
124 119 128	197 186 180	128 194 255				

Values of different variables as described earlier are given by:

x = 91.8889 45.0000 19.5556
y = 159.7778 89.1111 43.3333
z = 152.0000 138.7778 130.5556
w = 134.5556 90.9630 64.4815
v = 96.6667
C = 96
P = 93 144 199
$Z1$ = 145

So the average ROI and POI obtained after analysis are 96 and 145 respectively.

TABLE 4

Output excel file

Fig. input	*x*	*y*	*C*	*Z*1	Year	Month	Day	Hour	Minute	Second
1	13	239	44	38	2014	3	11	12	04	10
2	21	233	53	58	2014	3	11	12	24	19
3	29	228	96	145	2014	3	11	12	44	54

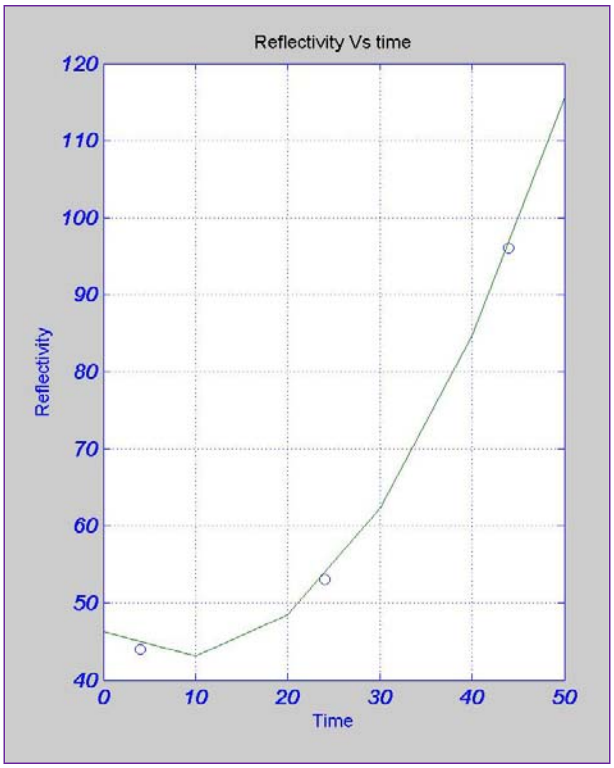


Fig. 12. LSA curve fitting of C *vs* Time

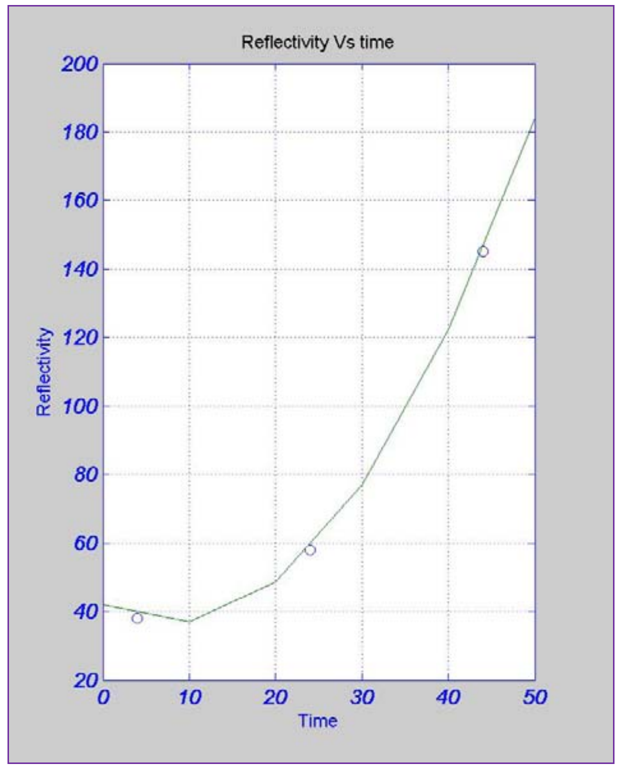


Fig. 13. LSA curve fitting of Z1 *vs* Time

4.1.3. *Extraction of co-ordinates and time information*

To extract the co-ordinates of POI from ROI, one program is developed in MATLAB (Appendix B) to display and extract the co-ordinate by mouse clicking. Similarly the Time of click can be stored in suitable format.

The output of the software after operated on specific POI is:

For ROI (Fig. 4) - *x*: 13.0, *y*: 239.0

For ROI (Fig. 6) - *x*: 21.0, *y*: 233.0

For ROI (Fig. 8) - *x*: 29.0, *y*: 228.0

All these above mentioned value can be displayed in a dialogue box immediately after clicking on POI.

The time of click can be displayed in the form of "Year, Month, Date, Hour, Minute, Second" as shown below:

>> C1 = 2014 3 11 12 04 10

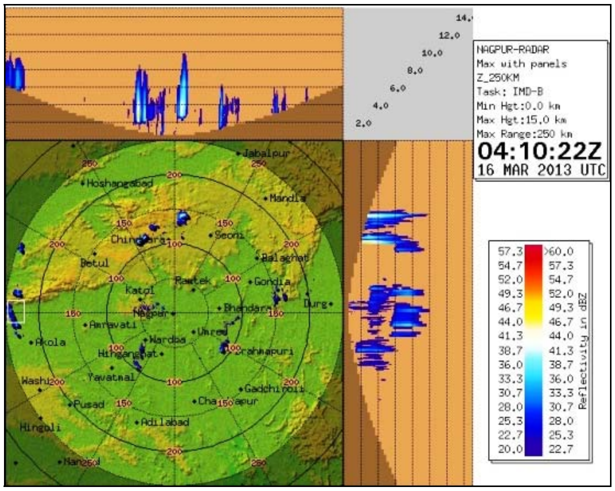


Fig.14. Cluster inside the rectangular shed with Actual Reflectivity:28dBZ

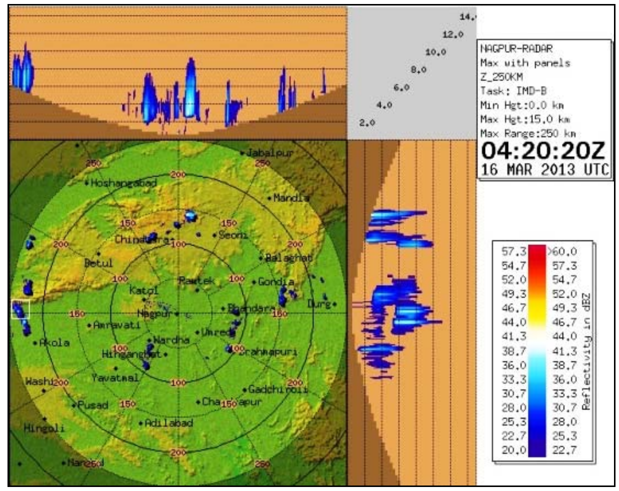


Fig. 15. Cluster inside the rectangular shed with Actual Reflectivity:35dBZ

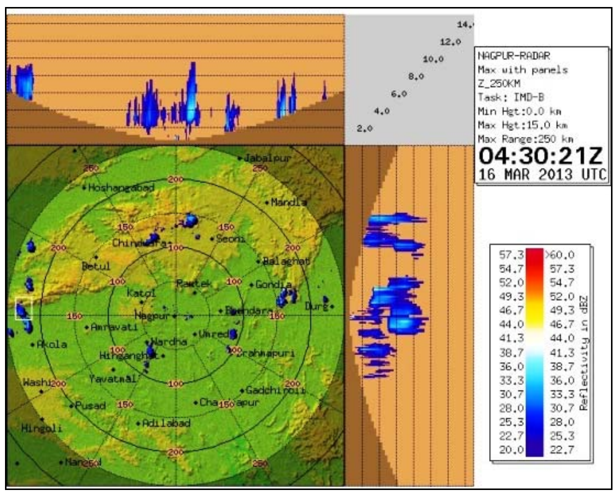


Fig. 16. Cluster inside the rectangular shed with Actual Reflectivity:30dBZ

Table 4 shows the Same Cloud Growth Statistics *i.e.*, various necessary data of same cloud cluster growth are extracted and stored in suitable format, *i.e.*, in an excel sheet for further analysis.

From Table 4 values like C and $Z1$ can be analyzed with respect to time to determine positive or negative cluster growth. Values of coordinates x and y can be used by PHDA to find out speed and direction.

Extraction of informations by the software from the excel sheet and the plotting its output is shown below. Output of C *vs* Time (Minute) and Z1 *vs* Time (Minute) is given below.This is based on best fit curve with Least Square Approximation (LSA) (Balgurusamy, 2008).

Figs. 12 &13 show the output of automatic curve fitting based on LSA (Least Square Approximation) where the values used and retrieved from the excel sheet as shown in Table 4. From the best fit curve it is verified that this cluster has a positive growth as that of actual cluster growth.

Similarly the output PHDA for speed and heading are:

Speed = 5.5 m/s and Heading = 88.66 deg

4.2. *Case 2*

Here to validate PHDA another case is analyzed with three different images of another growing cluster is taken. The data are as shown below:

Figs. (14-16) show the three cases of the same growing cluster at 10 Minutes interval where the ROI is the cluster inside the white rectangular shed. These images can be analyzed as described in case-1 and out puts are presented in tables 5, 6 and 7.

TABLE 5

ROI analysis for Fig. 14

*I*2(:,:,1)-Red	*I*2(:,:,2)-Green	*I*2(:,:,3)-Blue	*v*	*C*	P for POI	*Z*1
48 28 0 0 1 4 11 10 15 23	89 66 20 14 10 12 11 10 17 25	241 199 165 149 147 155 145 168 128 170	69.3667	69	4 12 155	57

Values of different variables as described earlier are given as:

$x =$ 15 13

$y =$ 29.4000 25.4000

$z =$ 165.2000 168.2000

$v =$ 69.3667

$C =$ 69

$P =$ 4 12 155

$Z1 =$ 57

So the average ROI and POI obtained after analysis are 69 and 57 respectively.

TABLE 6

ROI analysis for Fig. 15

*I*2(:,:,1)-Red	*I*2(:,:,2)-Green	*I*2(:,:,3)-Blue	*v*	*C*	P for POI	*Z*1
2 6 12 12 51 127 51 134 3 21 15 54	17 16 40 40 99 185 99 194 25 50 30 65	222 212 228 225 243 255 207 255 108 170 87 189	104.1389	104	134 194 255	194

Values of different variables as described earlier are given as:

$x =$ 22.3333 59.000

$y =$ 51.6667 91.6667

$z =$ 182.5000 217.6667

$v =$ 104.1389

$C =$ 104

$P =$ 134 194 255

$Z1 =$ 194

So the average ROI and POI obtained after analysis are 104 and 194 respectively.

TABLE 7

ROI analysis for Fig. 16

$I2(:,:,1)$-Red	$I2(:,:,2)$-Green	$I2(:,:,3)$-Blue	v	C	P for POI	$Z1$
4 15	23 34	203 250	90.4583	90	28 66 251	115
5 30	33 61	195 255				
0 28	33 66	174 251				
4 28	38 68	148 225				

Values of different variables as described earlier are given by:

x = 3.2500 25.2500
y = 31.7500 57.2500
z = 180.0000 245.2500
v = 90.4583
C = 90
P = 28 66 251
$Z1$ = 115

So the average ROI and POI obtained after analysis are 90 and 115 respectively.

TABLE 8

Output excel file

Fig. input	x	y	C	$Z1$	Year	Month	Day	Hour	Minute	Second
1	6	297	69	57	2014	3	18	14	10	36
2	9	294	104	197	2014	3	18	14	20	58
3	13	291	90	115	2014	3	18	14	30	51

Similarly co-ordinates and timing information's of the three POIs from ROIs can be obtained as below:

The output of the software after operated on specific POI is:

For ROI (Fig. 14) - x: 6.0, y: 297.0

For ROI (Fig. 15) - x: 9.0, y: 294.0

For ROI (Fig. 16) - x: 13.0, y: 291.0

These co-ordinates and time can be used by PHDA to find out speed and direction of specific cluster. All the extracted information such as reflectivity, time and co-ordinates can be stored in an excel file as shown in Table 8.

Table 8 shows the Same Cloud Growth Statistics *i.e.*, various necessary data of same cloud cluster growth are extracted and stored in suitable format, *i.e.*, in an excel sheet for further analysis.

From Table 8 values like C and $Z1$ can be analyzed with respect to time to determine positive or negative cluster growth. Values like x and y can be used by PHDA to find out speed and direction.

Extraction of informations by the software from the excel sheet and the plotting is presented. Output of C *vs* Time (Minute) and $Z1$ *vs* Time (Minute) is given in figs 17 and 18.

Figs. 17 and 18 are the output of automatic curve fitting based on LSA (Least Square Approximation) for case-2 where the values used and retrieved from the excel sheet are as shown in Table 8. From the best fit curve it is verified that this cluster has a negative growth as that of actual cluster growth.

Fig. 17. LSA curve fitting of C *vs* time

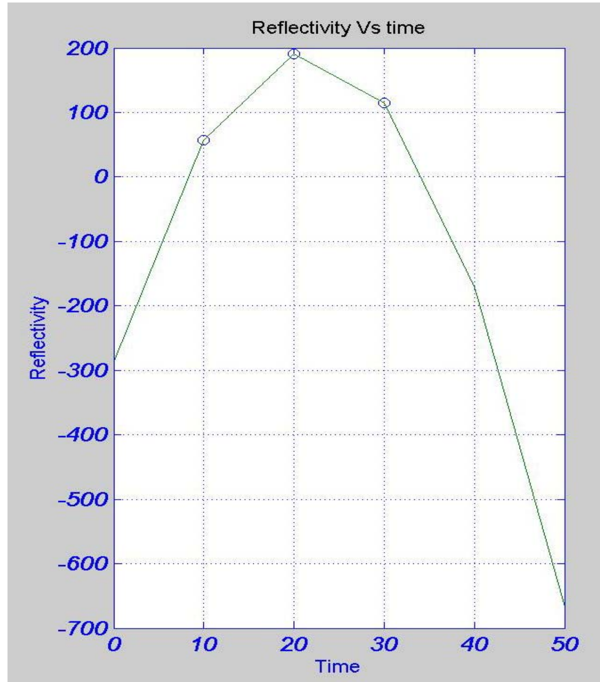


Fig. 18. LSA curve fitting of Z1 *vs* time

Similarly the output PHDA for speed and heading are:

Speed = 6.5 m/s

and

Heading = 89.44 deg

5. Conclusion

As PHDA gives the platform to estimate growth, speed and heading of a suspicious cluster, hence Table 3 & 8 are the final excel sheets/files with the stored information extracted from radar imageries. Two typical cases are considered for positive or negative cluster growth analysis. As extraction of actual reflectivity from imageries is not possible, hence only the cluster growth can be analyzed as positive or negative. The real growth of these known clusters is also found to be matching with our developed concept of radar image processing. Similarly from the excel file x, y co-ordinates and time can be retrieved by PHDA to find speed and heading of the cluster.

Acknowledgement

We are deeply grateful to ICAR (NICRA,) for funding this project entitled "Hailstorm Management Strategy in Agriculture". We are also thankful to MIT College of Engineering, Pune and Maharashtra, India. We are also grateful to Director General, India Meteorological Department for providing free access to radar data from various IMD Radar centers. Authors are particularly thankful to Mr. K. C. Sai Krishnan (Delhi) and Mr. S. G. Kamble (Mumbai) of India Meteorological Department (IMD) for promptly providing radar data.

References

Balgurusamy, E., 2008,"Numerical Methods", Tata McGraw-Hill, 24th Edition, ISBN-13:978-0-07-463311-3.

Blaschke, T., 2010, "Object based image analysis for remote sensing", *ISPRS Journal of Photogrammetry and Remote Sensing*, **65**, 2-16.

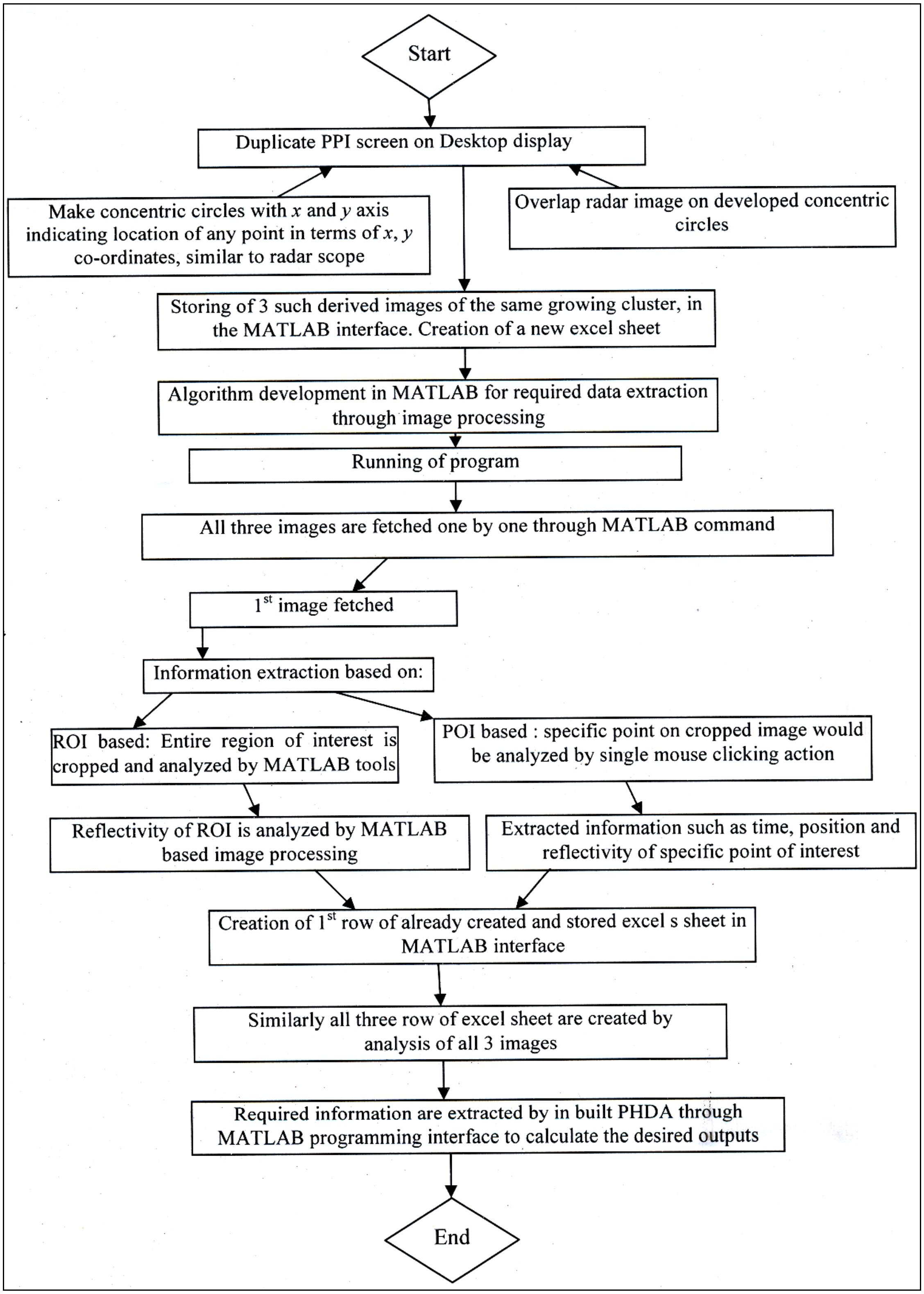
Gonzalez, R., Woods, R. and Eddins, S., 2012, "Digital Image Processing using MATLAB", Tata McGraw-Hill, Fifth Edition, ISBN-978-0-07-070262-2.

Kumar, P., 2010, "Hailstorm prediction, control and damage assessment", B S Publication, ISBN: 978-81-7800-248-4.

Mandhare, R., Upadhyay, P. and Gupta, S. 2013, "Pixel-level image fusion using brovey transform and wavelet transform", *International Journal of Advanced Research in Electrical*, Electronics and Instrumentation Engineering, **2**, 6.

"MATLAB @ Graphics", The Math Works, Inc. 3, Apple Hill Drive, Natick, MA 01760-2098.

Appendix A



Note : The entire process is done by a single MATLAB based software (refer Appendix-B) where in a single software PHDA is in-built which will automatically extract information for required outputs

Appendix B

MATLAB code for image loading and analysis

```
function demoOnImageClick
clc;clear;
%imObj = imread('ngp1.jpg');
%figure;
I = imread('ngp1.jpg');
figure
I2 = imcrop(I)
Mean(I2);
x=Mean(I2(:,:,1));
y=Mean(I2(:,:,2));
z=Mean(I2(:,:,3));
M=mean(z);
w=(x+y+z)/3;
v=mean(w);
C=fix(v);
D=fix(M);
%I2 = imcrop(I, rect);
%I2 = imcrop(I,[115, 68 130 112]);
imshow(I),colorbar,figure, imshow(I2);
grid on;
P = impixel(I2);
Z1=fix(Mean(P));
Cl=clock;
A1=fix(Cl)
filename='excel_final1.xlsx';
A = C;
B=Z1;
sheet=1;
%xlrange='E2';
SUCCESS = XLSWRITE(filename,A,sheet,'D2');
SUCCESS = XLSWRITE(filename,B,sheet,'E2');
SUCCESS = XLSWRITE(filename,A1,sheet,'F2:K2');
%I = imread('ngp1.jpg');
%figure;
%P = impixel(imObj);
%P = impixel(X,map)
%P = impixel(RGB)
hAxes = axes();
imageHandle = imshow(I);
set(imageHandle,'ButtonDownFcn',@ImageClickCallback);
function ImageClickCallback ( objectHandle , eventData )
axesHandle  = get(objectHandle,'Parent');
coordinates = get(axesHandle,'CurrentPoint');
coordinates = coordinates(1,1:2);
```

```
message     = sprintf('x: %.1f , y: %.1f',coordinates (1) ,coordinates (2));
helpdlg(message);
filename='excel_final1.xlsx';
A1=fix(coordinates);
sheet=1;
SUCCESS = XLSWRITE(filename,A1,sheet,'B2:C2')
helpdlg(message);
end
end
%xlswrite(filename,A,sheet,xlrange);
t=0:0.001:1;%Initializing time samples
%Transfer chara of sine and cosi.e sin Vs Cos will give circle
%Here we are plotting group of sine Vscos with differnt amplitudes
s1=3*sin(2*pi*t);%Sine wave with amplitude=3unit
c1=3*cos(2*pi*t);%Cosine wave with amplitude=3unit
s2=2*sin(2*pi*t);
c2=2*cos(2*pi*t);
s3=1*sin(2*pi*t);
c3=1*cos(2*pi*t);
plot(s1,c1,s2,c2,s3,c3);%Plotting sin Vs Cos
grid on;%Enable grid lines
axis equal;%Equal width of X and Y axis
```

MATLAB code for automatic data retrieving from excel file and displaying the outputs

```
clc;
clear all;
close all;
%t1=input('enter value of t1');
%t2=input('enter value of t2');
%t3=input('enter value of t3');
%x1=input('enter value of x1');
%y1=input('enter value of y1');
%x2=input('enter value of x2');
%y2=input('enter value of y2');
%x3=input('enter value of x3');
%y3=input('enter value of y3');

x = xlsread('excel_final1.xlsx',2,'J2:J4');
y = xlsread('excel_final1.xlsx',2,'E2:E4');
z = xlsread('excel_final1.xlsx',2,'D2:D4');
t3=xlsread('excel_final1.xlsx',2,'J4');
n=input('enter the degree of n');
p = polyfit(x,y,n);
%p = polyfit(X1,Y,n);
a=p(1,1);
b=p(1,2);
c=(p(1,3)-200);
%solve(a*x^2 + b*x + c == 0)
  X = zeros(2,1);

```

```
 d = sqrt(b^2 - 4*a*c);

 X(1) = ( -b + d ) / (2*a);
 X(2) = ( -b - d ) / (2*a);
 R1=(X(1)-t3)
 R2=(X(2)-t3)
p1= polyfit(x,z,n);
x2 = 0:10:50;
y2 = polyval(p,x2);
y3 = polyval(p1,x2);
subplot(1,2,1);
plot(x,z,'o',x2,y3)
xlabel('Time')
ylabel('Reflectivity (C)')
title('Reflectivity (C) Vs time')
grid on

subplot(1,2,2);
plot(x,y,'o',x2,y2)
xlabel('Time')
ylabel('Reflectivity (Z1)')
title('Reflectivity (Z1) Vs time')
grid on

% Create some values to plot
   %for i=1:20
   %  x(i) = i-10;
    % squared(i) = x(i) ^ 2;
     %cube(i) = x(i) ^ 3;
     %linear(i) = x(i);
     %log_of(i) = log(x(i));
     %sqrt_of(i) = sqrt(x(i));
   %end

   % Set the "grid" for plots to 2 rows and 3 columns. place
   % the first plot at location 1.

   %subplot(2,3,1);
   %plot(x,squared);
   %title('square');

   % place the last three plots across the second row
   %subplot(2,3,4);
   %plot(sqrt_of,cube);
   %title('sqrt');

   %subplot(2,3,5);
   %plot(linear,cube);
   %title('linear');

   %subplot(2,3,6);
   %plot(log_of,cube);
```

```
    %title('log');

    % place the cube plot (for instructional purposes) at location 3
    % (the upper right corner).

    %subplot(2,3,3);
    %plot(x,cube);
    %title('cube');
%X1 = xlsread('excel_final1.xlsx',2,'J2:J4');
%t1= xlsread('excel_final1.xlsx',2,'J4');
%Y = xlsread('excel_final1.xlsx',2,'E2:E4');
%n=input('enter the degree of n');
t1=xlsread('excel_final1.xlsx',2,'J2');
t2=xlsread('excel_final1.xlsx',2,'J3');
t3=xlsread('excel_final1.xlsx',2,'J4');
x1=xlsread('excel_final1.xlsx',2,'B2');
y1=xlsread('excel_final1.xlsx',2,'C2');
x2=xlsread('excel_final1.xlsx',2,'B3');
y2=xlsread('excel_final1.xlsx',2,'C3');
x3=xlsread('excel_final1.xlsx',2,'B4');
y3=xlsread('excel_final1.xlsx',2,'C4');


T1=(t1+t2)/2;
T2=(t2+t3)/2;
v1=(sqrt((x2-x1)^2-(y2-y1)^2))/(t2-t1);
v2=(sqrt((x3-x2)^2-(y3-y2)^2))/(t3-t2);
v3=(((v2-v1)/(T2-T1))*t3)+((v2*T1)-(v1*T2))/(T1-T2)
D=((y3/(x3-x1))+(y3/(x3-x2)));
D1=atan(D);
D2 = atand(D)
end
end
```