A weather radar data processing module for storm analysis
Ioannis N. Daliakopoulos and Ioannis K. Tsanis

ABSTRACT
A pre- and post-processing weather radar data module was developed in the Matlab suite of software with GIS data exchange abilities for storm event analysis. During pre-processing, each radar sweep is converted from spherical to Cartesian coordinates in the desired temporal and spatial resolution. The module’s functionality in post processing includes radar data display, geo-referencing over GIS maps, data filtering with the Wiener filter and single or multiple sweep processing. The user can perform individual storm cell detection and tracking, resulting in the storm’s average velocity and track length. The tested methods are modifications of the LoG (Laplacian of the Gaussian) blob detection method and a Brownian particle trajectory linking algorithm. Radar reflectivity factor (Z) data can be referenced over predefined rainfall (R) gauges in order to determine the radar Z–R equation parameters. The user can also produce spatially distributed precipitation estimates by using standard Z–R equations from the literature. The module’s functionality is demonstrated using data from a rainfall event captured by the NSA Souda Bay C-Band radar during a storm in October 2006. Results show that the Rosenfeld Tropical Z–R equation is the one that gives a satisfactory description of the spatial and temporal precipitation distribution of the investigated event.

Key words | detection, filters, Matlab, precipitation, radar, tracking

INTRODUCTION
Precipitation intensity can be estimated using ground precipitation gauges or a combination of gauges and radar data. Even though measurement errors such as undercatch (Adam & Lettenmaier 2003; Daliakopoulos et al. 2006) do exist, the precision of precipitation gauges is generally adequate and the derived ground level precipitation product is reliable. Nevertheless, coverage is often limited by topography and cost (Borga et al. 2000). Other effects such as the gradient of precipitation with altitude (Naoum & Tsanis 2004) can cause conventional interpolation of the precipitation variable to inadequately represent reality. Therefore, precipitation data can be insufficient for use with a distributed hydrological model. Weather radars have thus become an invaluable tool for the nowcasting of precipitation, that, besides their limitations (Delrieu et al. 2009), provide detailed spatial and temporal information (Divjak et al. 1999). Figure 1 shows the basic operation principles of weather radars from target detection to data recording and display using plan position indicator (PPI).

To make full use of the inherent radar capabilities, it is necessary to know the main sources of radar errors and limitations in order to properly interpret the data. Radar-rainfall error sources have been recognized and discussed in the literature for more than three decades (Harrold et al. 1974; Wilson & Brandes 1979; Zawadzki 1984; Austin 1987; Joss & Waldvogel 1990; Kitchen & Jackson 1993; Joss & Lee 1995). Michelson et al. (2005) present a comprehensive list of error sources that can be related to various factors from environmental conditions to radar hardware and model uncertainty. Several types of data corrections can be applied by the radar signal processor. For example, the $R^2$ correction normalizes the reflectivity with respect to distance from the radar transmitter, the signal degeneration correction calculates the degeneration due to atmospheric gases...
and clutter filtering represses the signal under a given threshold accounting for undesirable objects. The corrected information can contain some or all of the possible corrections but even after the corrections are made, various artefacts can still appear in radar data fields in several forms of noise (Divjak et al. 1999).

Noise introduces sharp gradients in otherwise smooth weather fields and can be eliminated, to some extent, by suitable texture smoothing. However, the smoothing process affects weather data, resulting in a loss of fine-scale details (Divjak et al. 1999). When noise is largely caused by a number of small sources, the system and observation noise can be regarded as a white Gaussian distribution (Maybeck 1993). Wilk & Gray (1970) applied this algorithm to data from the WSR-57 radar for the estimation of storm motion and precipitation. This technique and variations were applied by Zittel (1976), Brady et al. (1978), Crane (1979), Rosenfeld (1987) and Blanchet et al. (1991).

Over the past 40 years many algorithms have been developed for storm tracking (Johnson et al. 1998). Commonly, spatial cross correlation is used to determine the movement of storms (Bellon & Austin 1984; Austin 1987; Einfalt et al. 1990; Fabry et al. 1994; Tsanis et al. 2002). When applied to larger fields of reflectivity, this algorithm can provide accurate speed and direction information. According to a more recent technique, centroid identification and tracking can capture individual, isolated storms more effectively (Jackson 1993). Wilk & Gray (1970) applied this algorithm to data from the WSR-57 radar for the estimation of storm motion and precipitation. This technique and variations were applied by Zittel (1976), Brady et al. (1978), Crane (1979), Rosenfeld (1987) and Blanchet et al. (1991).

Mecklenburg et al. (2002), among others, present a comprehensive list of quantitative precipitation forecast (QPF) methods and tools. According to various sources, radar data manipulation software includes, but is not limited to the following.

1. EDGE (EEC 2011) is a state-of-the-art commercial license software developed by the Enterprise Electronics Corporation (EEC) and designed to work only with EEC’s radar output. This software supports a large number of data displays and corrections etc. but there is no export capability with GIS software.
2. The GFS forecasting algorithm, described by Toussaint et al. (2000a), uses cloud cell tracking or TREC/
COTREC motion field tracking in order to compute motion vector over a series of radar images.

3. DUR-TOOLKIT, described in Toussaint et al. (2000b), deals with visualization, pre- and post-processing and filtering. It is developed in C++ and therefore has to be portable; nevertheless there is no information about storm tracking or GIS interoperability.

4. Abacus, (Athanasiadis et al. 2009) has been developed as a radar data management and decision support system covering visualization, statistical estimations and weather conditions assessments. It is mainly a management platform for warnings etc.

The objective of the current paper is to develop a module that can serve as a workbench for scientific testing, experimenting and visualization allowing for easy integration of innovative functionalities for weather radar data processing. These functionalities include data pre-processing, filtering and visualization as well as storm detection and tracking. New algorithms on noise filtering, storm cell detection and tracking can be easily added to the module. Finally, the module allows for data export to GIS for further processing and visualization.

**METHODOLOGY**

The weather radar records the data in the form of volumes. A volume is constituted from a set of sweeps which in turn are made up from a set of beams (rays). Each beam consists of a number of range gates which represents the sampling resolution along the length of each beam. Each gate is the integration of the radar pulse for a particular distance that depends on the system configuration. The complete beam range covered by the radar is the product of the number of gates and the length of each gate.

As with most environment data, graphical display is an indispensable tool when seeking patterns, generating hypotheses and assessing the fit of proposed models (Tsanis & Gad 2001). The PPI (Figure 1) is the most common type of radar display. As the radar antenna rotates, a radial trace sweeps around it so the distance from it and the height above ground can be drawn as concentric circles. A simple data display such as this is often insufficient as a high degree of uncertainty affects precipitation estimates based on radar measurements (Anagnostou et al. 1999). Alternative displays include the constant altitude plan position indicator which gives a horizontal cross-section of data at constant altitude, vertical composite which produces images of the maximum reflectivity in a layer above ground, and others, according to the need of the users.

**Pre-processing**

The radar data are originally stored in polar coordinates and have to be converted to Cartesian coordinates in order to be displayed and processed. Along a single beam, the radar records measurements based on radar elevation angle $\theta_s$ and slant range $r$. The distance $s$ covered by the beam along the earth’s surface is given by (Doviak & Zrnić 1993):

$$s = k_s a \sin^{-1} \left( \frac{r \cos \theta_s}{k_s a + h} \right)$$

where $h$ is the height of the center of the radar beam given by:

$$h = \left[ r^2 + (k_s a)^2 + 2rh \sin \theta_s \right]^{1/2} - k_s a$$

In both equations, $a$ is the earth’s radius and $k_s$ is a multiplier which depends on atmospheric conditions. Assuming standard atmosphere, $k_s$ is equal to 4/3 (Doviak & Zrnić 1993). More complex equations like the ones presented by Gao et al. (2006) take into account the influence of thermographic profiles along the path of the radar beam.

For a complete sweep of the horizon at a single radar elevation $\theta_s$, data are represented by a set of polar coordinates ($s, \theta_s$). The corresponding Cartesian $x, y$ coordinates can be estimated using:

$$x = s \cos(\theta_s)$$

$$y = s \sin(\theta_s)$$

This process produces a scatter of ($x, y$) pairs. In order to store and display this data in an efficient way compatible
with most GIS applications, the data can be converted to an equally spaced grid form. At useful resolutions, that is, over 500 × 500 m, the disk storage required for a rasterized radar image is several orders of magnitude smaller than the raw radar data product and the produced image is several orders of magnitude smaller than the raw data file can be easier to handle and process. Unfortunately, rasterizing normally involves interpolating the data to a grid with fixed cell size which inherently degrades the original information. The extent of the grid used to interpolate data was chosen to be the horizontal projection of the maximum radius in the lower sweep of each volume. Finding a reliable interpolation scheme, especially for unevenly distributed data, represents a great challenge with the derived data seldom conveying additional information. In the case of weather radars, missing data signifies the absence of atmospheric phenomena and therefore estimating interpolated values at all costs can be meaningless, as suggested by Djurcilov & Pang (1999). Here, instead of a computationally expensive method like kriging (Simpson et al. 1998; Djurcilov & Pang 1999), the nearest neighbor (Goovaerts 1997) interpolation technique is used.

**Noise removal – the Wiener filter**

After interpolation, radar data images may contain noise. A digital signal \( s \) can deteriorate due to noise \( n \) with the resulting signal \( \hat{s} \) often modeled as a simple summation \( \hat{s} = s + n \). This noise can appear in several forms. In salt and pepper or speckle noise a small number of image pixels show a great discrepancy in color and intensity from their neighbors as a result of random fluctuations in the return signal from objects that are smaller than the radar image-processing resolution (Simontet 1970). The term salt and pepper originates from the black and white pixels that corrupt the image when viewed in monochrome. Gaussian noise, on the other hand, usually causes small changes in the original pixel values with the amount of distortion versus the occurrence frequency being normally distributed. Gaussian distribution is adopted assuming a sufficiently large number of pixels and noise for each pixel as an independent random variable.

Methods of de-noising radar fields could include convolving the original data with a low-pass or smoothing filter. De-noising \( s \) without a prior knowledge of its components requires making assumptions of the noise and signal characteristics. In general, radar measurement errors are not Gaussian (Krajewski & Ciach 2004). For a linear approach known as the Wiener filter (Wiener 1949), the procedure involves designing a filter \( h[x] \) such that:

\[
s[x] = h[x] \hat{s}[x] = h[x](s[x] + n[x])
\]

so that when the filter is convolved with the corrupted signal, the original signal can be recovered. Then, this constraint is transferred in the frequency domain and a quadratic error functional \( E \) is constructed:

\[
E(H(\omega)) = \int d\omega |H(\omega)(S(\omega) + N(\omega)) - S(\omega)|^2
\]

where \( \omega \) is the frequency parameter and capital symbols are used to denote the filter, the signal and the noise in the frequency domain, respectively. In order to simplify this expression the signal and noise are assumed to be statistically independent. To minimize, we differentiate and set equal to zero:

\[
H(\omega) = \frac{S^2(\omega)}{S^2(\omega) + N^2(\omega)}
\]

The process is described in detail by Farid (2008). Intuitively, this frequency response makes sense as when the signal is significantly stronger than the noise the response is close to 1, that is, the frequencies are passed. On the other hand, when the signal is significantly weaker than the noise the response is close to 0, that is, the frequencies are stopped. Assumptions about the statistical nature of the signal and noise are also necessary. For example a common choice is to assume white noise, \( N(\omega) \) is constant for all \( \omega \), and, for natural images, to assume that \( S(\omega) = 1/\omega^p \). Unfortunately denoising has the expected side effect of losing some of the image sharpness, as the Wiener filter is a low-pass filter.

In Matlab, the `wiener2` function is an adaptive application of the Wiener filter that estimates the local mean and variance around each pixel after Lim (1990):

\[
\mu = \frac{1}{NM} \sum_{n_1,n_2 \in R} \alpha(n_1, n_2)
\]
\[
\sigma^2 = \frac{1}{NM} \sum_{n_1,n_2 \in \eta} \alpha^2(n_1,n_2) - \mu^2
\]  

(9)

where \( n \) is the \( N \)-by-\( M \) local neighborhood of each pixel in the image. Then \texttt{wiener2} creates a pixel-wise Wiener filter using these estimates:

\[
b(n_1,n_2) = \mu + \frac{\sigma^2 - \nu^2}{\sigma^2(\alpha(n_1,n_2) - \mu)}
\]  

(10)

where \( \nu^2 \) is the noise variance. If the noise variance is not given, \texttt{wiener2} uses the average of all the local estimated variances. Figure 2 shows the effect of a \( 3 \times 3 \) Wiener filter on radar data.

Besides conventional image filters, more complex techniques such as neural networks (Teschl et al. 2007) and support vector machine (Ziyang et al. 2008) have been used to de-noise radar data, but are beyond the scope of the current paper.

\textbf{Z–R conversion}

The quantity that is measured by the radar is reflected energy called reflectivity which depends upon the size, shape, aspect, and dielectric properties of targets in atmosphere. Reflectivity Factor \( Z \) is measured in \( \text{mm}^6/\text{m}^3 \) and is a function of the number and size of drops within a given volume. The values of the reflectivity factor cover a wide range so they are commonly expressed in units of \( \text{dbZ} \). The radar rainfall rate can be obtained by using an empirical \( Z–R \) relation either from literature or deduced from measurements of drop-size distributions in natural rain (Battan 1973). The \( Z–R \) relation has the form:

\[
Z = aR^b
\]  

(11)

where \( a \) and \( b \) are empirical coefficients that depend on the type of precipitation (snow, rain, convective or stratiform) and geographic location (Austin 1987).

As the meteorological radar does not measure precipitation directly, error sources can influence the accuracy and precision of the estimates (Zawadzki 1984). The \( Z–R \) relation is non-linear and can vary depending on the geographic region and the event of rainfall to be estimated, which makes the correct choice of parameters complex (Battan 1973). The differences in radar estimates and terrain rain gauge measurements can in certain cases reach 100\% (Wilson & Brandes 1979) or larger. The sensitivity analysis

\[Figure 2\] Wiener filter application on radar data. Raw reflectivity factor data (left) is processed with a \( 3 \times 3 \) pixels Wiener filter and a 15 \( \text{dbZ} \) cut-off filter (right).
of precipitation estimates from the radar data of OSF (operational support facility) and the ‘WSR-88D Adaptive Parameter Working Group’ showed that the choice of valid Z–R relation can provide the most important improvement in the precipitation estimates (Belville 1999). This shows the importance of calibrating a suitable Z–R relation for the calculation of rainfall.

**Storm cell detection**

Techniques involving pattern recognition and image processing (e.g. Blackmer et al. 1973, Einfalt et al. 1990) have successfully been supplied to describe and recognize storm cells. Here, a method known as blob detection is being used in order to detect the storm cell centers. One of the first and most common blob detection techniques is based on the Laplacian of the Gaussian (Lindeberg 1998). A given image \( f(x,y) \) can be convolved by a Gaussian kernel \( \hat{h}_g \) of width \( \sigma \)²:

\[
h_g(x,y,\sigma) = \frac{1}{2\pi\sigma^2}e^{-(x^2+y^2)/2\sigma^2}
\]  

(12)

to give a representation:

\[
L(x,y,\sigma) = \hat{h}_g(x,y,\sigma) \cdot f(x,y)
\]

(13)

Then, the Laplacian operator is computed as:

\[
\nabla^2 L = L_{xx} + L_{yy}
\]

(14)

which usually results in strong positive responses of coherent reflective maxima of extent \( \sigma \) (Wildenauer et al. 2007). Essentially, the image \( f(x,y) \) is processed with a filter given by:

\[
h(x,y,\sigma) = \frac{(x^2 + y^2 - 2\sigma^2)\hat{h}_g(x,y,\sigma)}{2\pi\sigma^6 \sum_x \sum_y \hat{h}_g(x,y,\sigma)}
\]

(15)

The characteristics of the Gaussian kernel can be used as search criteria for the storm cells of different diameter and intensity.

**Storm cell tracking**

Having located dominant storm cell centers in a sequence of radar products, cell locations are matched up with corresponding locations in later frames to produce the trajectories in \( \rho(t, t) \). This requires determining which storm cell in a given frame is the most likely to match the one appearing in the adjacent frame. Tracking more than one storm cell requires defining the most probable set of \( N \) identifications between \( N \) locations in two consecutive frames. If the cells are indistinguishable, as for most storm cells, this likelihood can be estimated only by proximity in the two images. The corresponding algorithm for trajectory linking can be initiated by considering the dynamics of non-interacting Brownian particles as described in detail by Crocker & Grier (1996). For a given storm cell moving on a plane, the probability that it travels a distance \( \delta \) in time \( \tau \) is:

\[
P(\delta|\tau) = \frac{1}{4\pi D\tau} \exp \left( -\frac{\delta^2}{4D\tau} \right)
\]

(16)

where \( D \) is the diffusivity coefficient. Respectively, for \( N \) non-interacting identical particles the probability distribution can be derived:

\[
P(\{\delta_i\}|\tau) = \left( \frac{1}{4\pi D\tau} \right)^N \exp \left( -\frac{\sum_{i=1}^N \delta_i^2}{4D\tau} \right)
\]

(17)

The most likely storm cell trajectory from one frame to the next is the one which maximizes \( P(\{\delta_i\}|\tau) \) or, equivalently, minimizes \( \sum_{i=1}^N \delta_i^2 \). This criterion has been shown to perform well even for interacting cells provided a sufficiently small time interval between frames (Crocker & Grier 1996). Figure 3 shows an example of storm cell detection (nodes) and tracking (solid arrows). In order to account for storm cells that are not detected in all frames, their last known location is stored and matched with unassigned cells that appear in subsequent frames and fit the distance criterion.

Detection and tracking methodologies with similar algorithms, like CELLTRACK and COTREC are described by Kyznarová & Novák (2009).
An outline of the structure of the module is presented in Figure 4. The raw radar data are inputted to the module which initially converts them to a native format for subsequent read/write access. The module is comprised of a total of five interfaces that allow the user to manipulate the data and produce output.

The only currently available display mode is the PPI (Figure 1), which is the most common type of radar data display. The transmitting radar is usually placed in the center of the display so equal distances from it can be drawn in concentric circles. As the radar is revolving, the PPI trace appears to scan concentrically from the center to the largest distance of emission. North is found at the top of the display while the signal depicts the reflectivity at a single radar elevation. Therefore, it is possible to have one PPI display for each elevation scan.

The Filter selection menu gives the user the ability to choose among available preprocessing options for a selected object. Filters are modular so more options can be added in future versions. The currently available options are a 15 db threshold, an application of the noise reduction Wiener filter, a combination of the above and no filtering at all.

Depending on the season, geographic location, and expected weather type some standard Z–R relationships can be used to translate reflectivity into precipitation rate. Table 1 lists the Z–R relationships currently available along with the WSR-88D OSF recommendations for selecting the best Z–R relationship for most types of precipitation events (Belville 1999).

When mixed precipitation types are present the WSR-88D OSF suggests that sites should select a Z–R relationship based on the dominant type of precipitation. The use of a unique Z–R equation for all reflectivity observations regardless of differences in rainfall drop size distributions and atmospheric conditions can result in misinterpretation of the data (Atlas et al. 1999). Optionally,
Finally, the resulting in alternative storm dynamics representations. Parameters adjust the minimum local maximum intensity of the detected cells, change blob detection parameters in order to enhance the process. Parameters adjust the minimum diameter and the change tracking parameters in order to enhance the process. Parameters adjust the maximum distance between two storm cell instances in two consecutive radar scans.

The above operations are divided in two main layers that deal with: (a) operations of single radar scans, and (b) batch operations. The output can be either shown on the screen or saved in various formats such as common raster files (.bmp, .jpg, etc.) or geo-referenced raster files as Geotiff (Burrows 2000) and arcgrid (ESRI 1992). Geotiff and arcgrid formats can be handled by ESRI applications such as ArcMap, making display along with other geo-referenced layers as well as spatial calculations an easy procedure. Additionally, some objects like storm cell centers and storm tracks can be exported in the form of shape files which are also compatible with a wide range of GIS applications.

**CASE STUDY**

The module’s functionality is demonstrated using data from one rainfall event that was captured by the NSA Souda Bay C-Band radar during an October 2006 storm. The Radar of Souda Bay Naval Base is a model SWR-250C C-Band Radar constructed by EEC (Enterprise Electronics Corporation). The acquired data are further processed by the computer workstation software to generate and display a variety of weather products. The radar has been set up to perform five sweeps with elevations from 0.5 to 14°. The radar radiates a beam every 0.95° of rotation thus producing 378 rays of data for each of the five sweeps. With 239 gates per beam and a gate length of 1,000 m the maximum range of the radar is 239 km. Due to effects such as overshooting associated with the transmitter elevation, partial beam filling that can cause reflectivity factor and precipitation rate underestimation (e.g. Joss & Waldvogel 1990; Durden et al. 1998) and beam attenuation (e.g. Paulitsch et al. 2009; Cremonini & Bechini 2010), quantitative precipitation estimation at ranges beyond 100 km is problematic (Uijlenhoet et al. 2006).

In October 2006, a frontal depression moved eastward towards the central Mediterranean and crossed the island of Crete at midday on October 17. This depression caused a high-intensity short-duration heavy rainfall resulting in a flash flood event in the Almyrida basin, a 25 km² watershed located in the northwest part of the island. At the time of the event the neighboring rain gauge of Souda Bay (16 km) recorded a maximum hourly precipitation of 25.2 mm and a daily gauge located just 3 km from the watershed recorded 220 mm. Similar to other intense precipitation events in the Mediterranean (Berne et al. 2009), the flash flood was devastating, leading to the loss of one life and over €1M in damages in Almyrida alone, and leaving a total damage toll of approximately €3M. With the help of radar data it was possible to reconstruct the event and identify characteristics of the storm such as storm cell velocity and precipitation intensity. The developed module can process radar data and export them to GIS for a better visualization and understanding of the meteorology of the event. This is more efficient than building the code within a GIS system which is slow and not as versatile (Naoum & Tsanis 2003).

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Typical Z-R relationships including the phenomena for which their use is recommended in literature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship</td>
<td>Optimum for</td>
</tr>
<tr>
<td>Marshall-Palmer ($Z = 200R^{1.6}$)</td>
<td>General stratiform precipitation</td>
</tr>
<tr>
<td>East-Cool Stratifrom ($Z = 130R^{2.0}$)</td>
<td>Winter stratiform precipitation – east of continental divide</td>
</tr>
<tr>
<td>West-Cool Stratiform ($Z = 75R^{2.0}$)</td>
<td>Winter stratiform precipitation – west of continental divide</td>
</tr>
<tr>
<td>WSR-88D Convective ($Z = 300R^{1.8}$)</td>
<td>Summer deep convection</td>
</tr>
<tr>
<td>Rosenfeld Tropical ($Z = 250R^{1.2}$)</td>
<td>Tropical convective systems</td>
</tr>
</tbody>
</table>
On the day of the event, raw radar data were acquired at 15-min intervals except for several missing scenes due to power outages in the area. Figure 5 shows the reflectivity recorded shortly after the formation has crossed over the island passing through two mountaintops with a northward direction. Data were interpolated to a 500 m × 500 m Cartesian grid providing adequate resolution for further processing. The observations were filtered using a Wiener filter coupled with a 15 db threshold to remove noise and insignificant reflectivity values. Then, it became clear that

---

Figure 5 | Reflectivity of the storm that hit the north coast of Crete on October 17, 2006 as it was recorded on 15:13 UTC. Arrows show the direction of individual storm cells from 15:13 to 15:58 UTC.
the system was essentially a large cyclone with a maximum diameter of about 200 km. Storm cell detection and tracking produced vectors that were exported to GIS along with reflectivity values. Vectors depicted individual storm cell movement within the formation during the hours of the storm, showing the cyclone's center persisting to the northeast of Almyrida, a position that favors orographic uplift from the mountain volumes in the south of the watershed.

Table 2 shows rainfall rates versus dbZ for all cited models. For reference, the maximum reflectivity factor recorded over Almyrida at 15:13 UTC of October 17, 2006 is included for reference.

<table>
<thead>
<tr>
<th>dbZ</th>
<th>Marshall-Palmer</th>
<th>East-Cool Stratiform</th>
<th>West-Cool Stratiform</th>
<th>WSR-88D Convective</th>
<th>Rosenfeld Tropical</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.00</td>
<td>0.32</td>
<td>0.49</td>
<td>0.65</td>
<td>0.20</td>
<td>0.18</td>
</tr>
<tr>
<td>25.00</td>
<td>1.33</td>
<td>1.56</td>
<td>2.05</td>
<td>1.04</td>
<td>1.22</td>
</tr>
<tr>
<td>35.00</td>
<td>5.62</td>
<td>4.93</td>
<td>6.49</td>
<td>5.38</td>
<td>8.29</td>
</tr>
<tr>
<td>40.33</td>
<td>12.09</td>
<td>9.11</td>
<td>11.99</td>
<td>12.92</td>
<td>23.04</td>
</tr>
<tr>
<td>45.00</td>
<td>23.68</td>
<td>15.60</td>
<td>20.53</td>
<td>27.86</td>
<td>56.46</td>
</tr>
<tr>
<td>55.00</td>
<td>99.85</td>
<td>49.32</td>
<td>64.93</td>
<td>144.28</td>
<td>384.64</td>
</tr>
</tbody>
</table>

Table 2 | Rainfall rate comparison (values in mm/h). The maximum reflectivity factor recorded over Almyrida at 15:13 UTC of October 17, 2006 is included for reference.

A maximum precipitation rate of 23 mm/h is in good agreement with the field data. This rate is estimated offshore from Almyrida and can be tracked back over the watershed during the time of the flood. Other models estimate lower rainfall rates giving a less representative image of the rain fields for this particular storm. At the time of the study, the validation or calibration of a custom Z-R relationship was not possible as available terrain gauge data have inadequate temporal and spatial resolution. Nevertheless, the results showed the spatial and temporal distribution of the formation that caused the flash flood which by itself proved the usefulness of this module in the study of extreme events via analysis of weather precipitation radar data.

CONCLUSION

A new module for weather radar data pre- and post-processing was developed in Matlab. This paper briefly presents its functionalities with respect to data analysis and visualization. The tool allows for modularity, therefore serving as a workbench for the comparison of different algorithms. Each of the operations performed (i.e. filtering, storm cell detection, etc.), can be executed using alternative

![Figure 6](https://iwaponline.com/jhi/article-pdf/14/2/332/386685/332.pdf)
algorithms. Each new algorithm can be added as part of the module GUI in a new option of the corresponding drop down menu. This allows for quick testing and identification of the most appropriate methodology for each case. Results can be compared visually and mathematical computations can be performed in any arcgrid-compatible environment.

Regarding the default algorithms, each has its own advantages and limitations. For example, the storm cell tracking algorithm performs best with distinguishable storm cells having consistent paths. Irregular or rapid storm cell motion and ambiguous formations that merge and divide hinder the effectiveness of the algorithm. Nevertheless, using the algorithms in a modular environment allows for quick and efficient testing and result comparison for algorithm improvement and extension.

The module was tested in the study of a flash flood event in north-western Crete in an attempt to reconstruct the meteorological conditions that lead to its outbreak. The functionalities of data preprocessing, filtering and storm cell detection and tracking gave a good representation of the storm formation and movement. Particularly, the Rosenfeld Tropical Z–R equation provided an adequate fit with precipitation rate measurements during the event. Finally, the results were exported in a GIS-compatible format allowed for better visualization and easier manipulation in a friendly environment.

ACKNOWLEDGEMENTS

The research presented in this paper has been carried out as part of the European Commission FP6 funded project, HYDRATE. The authors also wish to thank Enterprise Electronics Corporation (EEC) for providing data and support throughout this research.

REFERENCES


