

Extreme Compression of Weather Radar Data

Vishnu V. Makkapati, *Member, IEEE*, and Pravas R. Mahapatra

Abstract—A method for achieving extreme levels of compression of high-volume weather radar data is presented. Weather reflectivity contours, as per National Weather Service or custom thresholds, are processed by tracing their departure from a smoothed version to obtain the local extrema which serve as control points. The control points, which are transmitted in relative coordinates for further compression, are interpolated using a second-degree B-spline to retrieve the contours. The encoding–decoding method is capable of capturing the random undulations inherent in weather contours. It is shown that over two orders of magnitude of compression is possible without perceptible loss of meteorological information. Multiple enhancements to the basic method are quantitatively studied and compared with the existing methods for radar data compression.

Index Terms—Contour-based data compression, extreme data compression, meteorological radar, weather radar data compression.

I. INTRODUCTION

OVER the recent decades, the Doppler weather radar (DWR) has emerged as a primary instrument for observation and dissemination of weather data. It offers continuous, high-resolution, accurate, and multiparameter observation capabilities over large geographical areas, which are of immense value for local as well as synoptic weather information systems. However, these very qualities result in the DWRs generating huge volumes of digital data that need to be processed, stored, and transmitted in real time. Archival, search, and retrieval of DWR data also pose serious problems because of its immense volume. All these operations would be substantially simplified by compressing the digital DWR data while retaining the significant meteorological information of the desired scales.

While compression of the DWR data is highly desirable in general, there are special situations where very high levels of compression are essential. One such application arises in the context of transmission of pictorial weather data between aircraft and ground. Weather is a common cause of flight delays and is a contributing factor for many accidents. The availability

of the current weather picture in the area of flight to the pilot helps her/him avoid or skirt zones of adverse weather. Because of the high susceptibility of aircraft to local and small-scale weather features, and the need for highly current information, DWR is the primary source of detailed graphic weather data for aviation [1]–[3]. Since much of the DWR observation, mosaicing [4], and interpretation are carried out using ground-based radars such as the U.S. NEXRAD (WSR-88D) network and the dedicated terminal DWRs at important airports, the data need to be uploaded through a ground–air link for display in the aircraft cockpit. In many situations (e.g., storm observation from aircraft), it is also necessary or desirable to download data from airborne weather radars to ground stations for interpretation, fusion with ground-based data, and archival for future investigation (e.g., in case of accidents or incidents).

Air–ground data links have severe bandwidth limitations, with safety- and flight-critical data (such as air-traffic-control (ATC) instructions) receiving the highest priority. Even in cases where spare link capacity may be available, the cost of such bandwidth is very high. This necessitates the compression of weather radar data to the maximum extent possible. Similar considerations apply for data transmission between ATC centers and for data dispersal over the Internet. Although these links usually provide higher bandwidths than the ground–air link, they are also required to carry more data; hence, high compression of individual data components is a priority. Although discussed here in the context of weather radar data, the compression technique will be directly applicable in many other areas such as efficient archiving and transmission of cartographic, geodetic, and geophysical data, particularly contour maps. Similarly, storage and transmission loads in bandwidth-intensive motion picture applications, particularly animation pictures (which are essentially contour based), can be grossly minimized by using this method.

Image compression methods exploit the spatial redundancy to achieve compression [5], [6]. These schemes can be broadly classified as lossless and lossy methods. Lossless methods [7], [8], by definition, are constrained to retain all the information in the original data. They can achieve only low to moderate degrees of compression in the case of unstructured (randomly varying) data fields such as weather pictures [9], [10]. Lossy methods exploit the psycho-visual redundancy and are permitted to sacrifice the less observable features of data fields to varying degrees to offer higher compression ratios [11], [12]. The JPEG and JPEG 2000 compression schemes are the global standard schemes for general-purpose compression [13], [14]. However, generic compression schemes that are not optimized for any particular class of data seldom yield compression ratios over an order of magnitude for weather pictures. Schemes that achieve significantly higher compression must take into account

Manuscript received March 21, 2006; revised March 13, 2007. This work was supported by Honeywell Technology Solutions Laboratory, India, and the publication is supported by Philips Research India–Bangalore. This work formed part of V. V. Makkapati's M.Sc. (Engg.) research at Indian Institute of Science, Bangalore.

V. V. Makkapati is with Philips Research Asia–Bangalore, Philips Electronics India Ltd., Philips Innovation Campus, Manyata Tech Park, Bangalore 560 045, India (e-mail: vishnu.makkapati@philips.com).

P. R. Mahapatra is with the Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560 012, India (e-mail: pravas@aero.iisc.ernet.in).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2007.903675

the structure of specific data fields and their intended application. We attempt here to devise high-compression schemes specifically tuned for weather radar reflectivity data.

As the reflectivity data are most commonly used in a contoured form, most methods for very high data compression rely on the efficient encoding of such contours [15]. Chain codes offer a convenient and efficient method of representing contours [16]. In this method, a point on the contour is represented relative to its neighboring point in clockwise or counterclockwise directions with the help of three bits. However, this scheme is highly susceptible to channel errors as any error, once introduced into a chain code value, propagates over the remaining length of the contour. Hence, schemes that represent the global shape of contours are preferable. Other methods proposed for general contour encoding may be found in [17] and [18].

In the specific context of weather radar data compression, the polygon-ellipse method proposed in [19] represents a reflectivity contour using straight-line or elliptical segments. The polygonal scheme designed in [20] approximates a contour using straight lines, the ends of which are termed vertices.

This paper presents a spline-based technique for the extreme compression of weather radar reflectivity data while retaining the meteorological features of the weather field that are significant for its intended application. Certain variants of the method to improve the performance of the basic method are also presented. This paper is organized as follows. Section II provides the background and terminology associated with weather radar data generation. Section III explains the principles of spline interpolation. The basic algorithm for the proposed scheme is presented in Section IV, and its performance is evaluated. Section V outlines certain enhancements to the basic method and estimates the resulting performance improvement. Comparisons are also made with prior compression methods. Conclusion of this paper is drawn in Section VI.

II. BACKGROUND

A radar emits a coherent train of microwave pulses along a directed beam and processes the reflected pulses or echoes to derive the parameters of the scatterers within its detection range [22]. The volume within the radar beam sampled at a given instant constitutes a resolution volume or bin. A radially contiguous set of bins is termed as a radial or ray. The angle of a radial with respect to a fixed reference (typically north) is called its azimuth. A weather radar beam generally scans circularly at a fixed elevation angle; a set of the successive rays accumulated at an elevation angle is called a sweep. Azimuthal scans are usually performed at a set of elevation angles, resulting in a volume scan. The choice of the elevation angles, the resolution of azimuth scan, and the number of bins depend on the type of radar used, the mode of operation, and the intended application.

Reflectivity is a scalar parameter indicating the strength of the echo returned from a given radar bin and is used as a measure of the intensity of the instantaneous precipitation rate at the location [3]. The reflectivity factor Z has a very large dynamic range, and hence, it is normally represented in decibels dBZ . A 2-D depiction of the reflectivity within the bins occurring in a circular scan, as obtained with a plan-

TABLE I
NWS REFLECTIVITY THRESHOLDS [1], [21]

Level	Reflectivity Interval dBZ	Rainfall Category
1	18 - 30	Light
2	30 - 41	Moderate
3	41 - 46	Heavy
4	46 - 50	Very heavy
5	50 - 57	Intense
6	> 57	Extreme

position-indicator (PPI) display, yields a visual picture of the rainfall intensity distribution in the radar field corresponding to the elevation of the scan. In such a display, the reflectivity values are generally color-coded according to intensity. On a black-and-white display, the coding is done in shades of gray.

Interpretation of reflectivity fields and the assessment of their hazard potential are often facilitated by using contoured displays delineating the regions of the field that cross-defined reflectivity thresholds signifying qualitatively different rainfall levels. A commonly used set of thresholds are those specified by the National Weather Service (NWS), as shown in Table I. Fig. 1 shows an actual PPI display of reflectivity, and its contours corresponding to the NWS reflectivity thresholds.

III. B-SPLINE INTERPOLATION

The B-spline is a generalization of the Bezier curve [23]. Let a knot vector be defined as $T = \{t_0, t_1, \dots, t_m\}$, where the constants $t_i \in [0, 1]$, called knot values, are strictly in nondecreasing order. The curve defined by a parameter t

$$C(t) = \sum_{i=0}^n P_i N_{i,p}(t) \quad (1)$$

is a B-spline. $C(t)$ denotes the location of the point on the curve corresponding to the parametric value t . The curve is drawn for various values of t , ranging from t_{\min} to t_{\max} , typically having values of zero and one, respectively. The shape of the B-spline curve can be controlled by $(n + 1)$ control points P_i in the object space. The positions of the control points are chosen to achieve the desired shape of the curve. The B-spline curve does not necessarily pass through its control points but always lies within the convex hull of these points (Fig. 2).

The degree p of the curve is defined as

$$p \equiv m - n - 1. \quad (2)$$

Increasing the degree p results in a smoother curve. The knot values $t_{p+1}, \dots, t_{m-p-1}$ are called internal knots. A uniform B-spline curve is one with equally spaced internal knots.

The basis function $N_{i,p}(t)$, which is recursively defined as

$$N_{i,0}(t) = \begin{cases} 1, & \text{if } t_i \leq t < t_{i+1} \text{ and } t_i < t_{i+1} \\ 0, & \text{otherwise} \end{cases}$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) \quad (3)$$

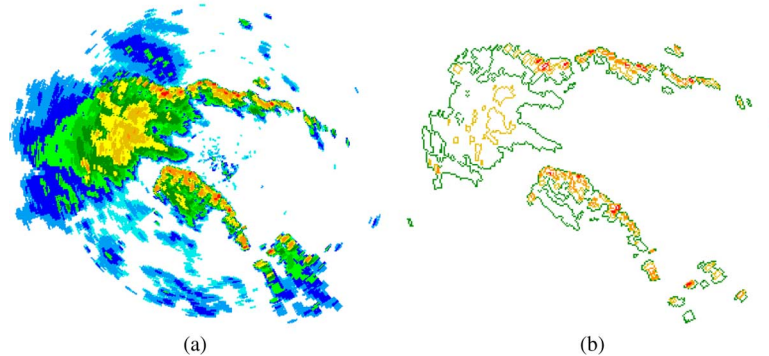


Fig. 1. Sample PPI reflectivity field from WSR-88D DWR in Level II format (elevation 0.35° , display 512×512 pixels with eight-bit depth). (a) Reflectivity distribution. (b) NWS reflectivity contours as per Table I.

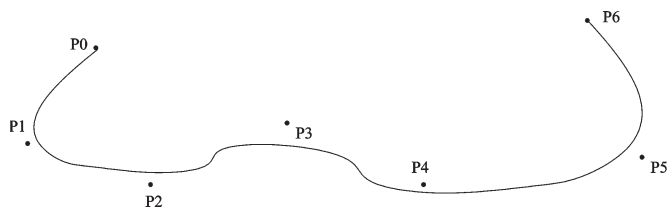


Fig. 2. Schematic B-spline representation of a curve.

determines the extent to which a given control point affects the curve for a parametric value t . $N_{i,p}(t)$ is a polynomial of degree p in t and is nonnegative for all i 's, p 's, and t 's.

The basis functions have a nonzero value only in the open interval $t \in (t_i, t_{i+p+1})$. The sum of the control points is an affine combination, where the sum of all the basis function values of the control points is equal to one. However, the latter sum does not equal unity in the intervals $[t_0, t_p]$ and $[t_{n+1}, t_{n+p+1}]$; no affine combination of the control points is possible in such cases. The B-spline curve is thus defined only in the interval $[t_p, t_{n+1}]$, and interpolation near the end points is not inherently possible. End point interpolation can be achieved by considering the first $p+1$ and last $p+1$ knot values to be equal to t_{\min} and t_{\max} , respectively. A curve is $p-k$ times differentiable at a point where k duplicate knot values occur.

The B-splines provide a common and convenient mathematical representation of standard analytic shapes as well as free-form curves and are affine invariant. Such a representation is chosen here for the following advantages. 1) Changes to a control point affects only the curve in its vicinity. 2) Any number of control points can be added without increasing the degree of the curve. 3) Adding multiple points at or near a position draws the curve toward that position. 4) Closed curves can be represented by making the first and last points the same. The B-splines of degree 2 or more result in smooth curves, but sharp corners can be introduced by joining two spline curve segments. The B-splines, which are polynomial curves, cannot represent many useful simple shapes such as circles and ellipses. Hence, a generalization of B-spline that is known as nonuniform rational B-splines has been designed.

IV. PROPOSED SCHEME

As weather reflectivity data are frequently utilized in a contoured form, it is logical to store and transmit them in such a format. This paper performs the generation, compression, and retrieval of contoured reflectivity maps through the following steps [24].

A. Contour Tracing

The reflectivity field is converted to a binary image by applying a threshold. A contour is then defined as a boundary that separates a region of zeros from a region of ones. A contour enclosing a region of ones is called a region contour, and a contour with only zeros inside is called a hole. The region and hole contours may be nested, i.e., there may be a region contour inside a hole contour, a hole contour inside a region contour, and so on.

We use a radial sweep algorithm [25] for tracing the contours in a binary image. However, this method does not automatically trace the internal structures of a contour, e.g., holes, region contours inside them, and nested contours. Further, it traces only one contour at a time, after a point on the contour has been identified using a search process. To overcome these drawbacks, the method proposed in [26] has been recursively applied to trace the nested hole and region contours. This scheme increments the value of the pixels on a contour by one while tracing it, i.e., it assigns a value of two to points on the contour. If there are any pixels common to a region contour and a hole contour, these are assigned a value of three. The direction of interior of a contour (relative to the traverse direction, which may be clockwise or counterclockwise) is determined using chain codes, and the pattern of three successive pixels is utilized to identify the region and hole contours.

B. Extraction of Control Points

A closed contour in two dimensions can be specified using the coordinates of its n points (x_i, y_i) , $1 \leq i \leq n$, and it satisfies the conditions: 1) $|x_i - x_{i+1}| \leq 1$ and $|y_i - y_{i+1}| \leq 1$; and 2) $(x_{n+i}, y_{n+i}) = (x_i, y_i)$. The goal here is to extract the

control points (x_i, y_i) , $1 \leq i \leq m$ that can faithfully represent a contour where $m \leq n$. The case $m = n$ corresponds to transmitting the original contour itself. A novel method of extracting the control points is developed here.

When trying to represent a contour by a relatively small number of points, the obvious criterion would be to choose the dominant points, i.e., those points which maximally retain the information contained in the contour. Attneave [27] made the profound observation that information on the shape of a curve is concentrated at points having high curvature. A large body of research and algorithms has been based on this criterion (e.g., [18] and [28]). The rationale for Attneave's criterion is that arc segments between points of high curvature would have relatively low curvature and, hence, can be approximated by straight-line segments while reconstructing the contour. Such an approach serves very well to describe the outlines of objects of daily life which are often relatively smooth. Weather contours, in contrast, generally present a highly jagged appearance, with random high-frequency undulations [Fig. 1(b)]. Determining local curvatures in such cases becomes extremely cumbersome. Since radar display frames, each with numerous contours, are generated every few seconds, real-time encoding of the contours requires simple and fast algorithms. It may be noted that other natural contours such as those derived from vegetation and topography are also fractally undulating, but their static nature lends them to computation-intensive encoding on a one-off basis.

Algorithms based on distance measure are expected to be computationally more tractable than those requiring curvature computation, particularly for rapidly and randomly undulating contours. This has been the basis of the previous methods for encoding weather radar contours [19], [20]. Our method is based on the premise that vertices signifying local extrema in specified scales naturally define the perceptually significant points in a contour [29]. Further, while high curvature points serve as good key points for straight-line reconstruction, our approach utilizes a spline-based reconstruction which permits curvilinear interpolation. Since a spline interpolation confines itself to the convex hull of the control points (Fig. 2), it will not track maximal points, if nonmaximal points are taken as control points. We therefore use the local extremum criterion in choosing the control points for encoding the contours.

Our scheme first obtains a smoothed contour by averaging the original contour over a certain neighborhood of each point as follows:

$$x_p^s = \frac{1}{L+1} \sum_{i=p-\frac{L}{2}}^{p+\frac{L}{2}} x_i \quad y_p^s = \frac{1}{L+1} \sum_{i=p-\frac{L}{2}}^{p+\frac{L}{2}} y_i \quad (4)$$

where (x_p, y_p) are the coordinates of the p th point on the original (unsmoothed) contour, $L+1$ is the number of points over which the contour is smoothed, and (x_p^s, y_p^s) are the coordinates of the p th point on the smoothed contour.

The smoothing length may be specified absolutely or as a fraction of the overall contour length. To start with, the scheme determines a crossing point between the original and smoothed contours by searching for a point where the distance

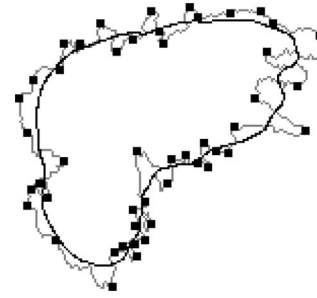


Fig. 3. (Lighter line) Raw contour and (darker line) its smoothed version. The small black squares are the control points.

between the two contours is one pixel or less. Then, a search is performed to find a point on the original contour with maximum deviation from the smoothed contour before the next crossing point is encountered. Such a point is designated as an extremal control point.

If the length of the segment between two crossing points exceeds a certain length, additional control points are added midway between the extremal control point and the crossing points on either side of it. This process is repeated until all the points on the original contour have been traversed, i.e., until the starting point has been reached. The scheme may not properly extract the control points for smooth and small contours (e.g., contours with less than 15 points). In such cases, uniformly spaced points on the original contour are chosen as control points. An example contour, its smoothed version, and the extracted control points are shown in Fig. 3.

Determination of the nearest point on the smoothed contour for a point on the original ("raw contour" in Fig. 3) is not a trivial problem since, in general, a one-to-one correspondence between the points in the original and smoothed contours does not exist. This is because the number of points in the smoothed contour is less than those on the original contour. A brute force search for the nearest point on the smoothed contour for each point on an original contour incurs significant computational cost. We minimize the computing effort by starting from a crossing point, considering successive points on the original contour, and confining the search for the nearest point on the smoothed contour to a neighborhood of the previous nearest point.

Since the control points constitute nearly the entire transmitted data, their number for a given contour determines the compression ratio. This ratio may be altered within limits by adjusting the number of control points. The control points may be prioritized based on their distance from the smoothed contour, with higher importance given to farther points. Elimination of the lowest priority points would enable achieving higher compression ratios with minimal loss of contour fidelity.

C. Contour Storage and Transmission

To store or transmit the control points, each point would require $\{\lceil \log_2(M) \rceil + \lceil \log_2(N) \rceil\}$ bits, where M and N denote the width and height of the input reflectivity field, respectively. Since control points constitute the maximum percentage of the data load, we reference them to the top left corner

(x_{\min}, y_{\min}) of the contour's minimum bounding rectangle (MBR) in order to save on the bit requirements. Then, each point would require $\{\lceil \log_2(x_{\max} - x_{\min}) \rceil + \lceil \log_2(y_{\max} - y_{\min}) \rceil\}$ bits, where (x_{\max}, y_{\max}) are the coordinates of the lower right corner of the MBR. The coordinates of the top left corner, however, have to be transmitted once for each contour.

The thresholds used in generating the contour representation also have to be transmitted to the receiving end. To compress this information, these thresholds are stored in a lookup table (LUT), and only the indices to this LUT are transmitted. In general, multiple closed contour curves would exist at any given threshold level, and these are numbered before transmission. A tag bit of zero and one has been used to distinguish between the region and hole contours, respectively.

D. Retrieval of Contoured Fields

The contour is reconstructed from the control points at the receiving end using a B-spline interpolation. Let $k_i + 1$ denote the number of control points received for the i th contour. The parameter t is incremented in steps of $1.0/(k \times (k_i + 1))$, where k is an integer. Higher values of k generate duplicate points because of rounding off to integer pixel values, which may be discarded. A low value for k may cause gaps between successive points. Through repeated numerical experiments, we have found a range of 2–10 to be appropriate for k and have adopted a “safe” value $k = 10$ which almost completely avoids gaps, although it generates some duplicate points. Any gaps still remaining between the successive reconstructed points are filled using Bresenham's line drawing algorithm [30].

The interior of each reconstructed contour has to be filled with the threshold value of the contour using a region-filling algorithm. The region contours for smaller thresholds are larger than those of the higher thresholds. However, the hole contours at higher thresholds are larger than those at smaller thresholds. An efficient scheme should fill these contours without any redrawing, i.e., only the area inside the region contours but outside the hole contours should be filled with the color corresponding to the given threshold. The proposed scheme first draws the region and hole contours at the lowest threshold value and the next higher threshold with the color corresponding to the lowest threshold. Then, an arbitrary point inside a region contour at the lowest threshold value is identified [26]. Similarly, an arbitrary point inside the hole contour at the next higher threshold is also obtained. These points are used as seed points for the boundary fill algorithm. This scheme is repeated for the next higher threshold and so on. A composite contoured image of the reflectivity is generated when all the thresholds are exhausted.

E. Performance Evaluation

The performance of a compression scheme may be evaluated using the resulting compression ratio and the fidelity of reconstruction. Compression ratio is defined here as the ratio of the number of bits required to store the raw reflectivity image to the number of bits required to transmit the contour-coded image. The fidelity may be evaluated quantitatively and qualitatively.

The metrics commonly used for quantitative assessment are mean-square error, root-mean-square error (rmse), signal-to-noise ratio, and peak-signal-to-noise ratio [31], [32]. We use the rmse between the original and reconstructed contours as a quantitative metric for evaluation of the scheme [33]. For qualitative assessment, a visual comparison is made between the original and reconstructed reflectivity fields. The results of spline interpolation are compared with those from other comparable methods such as the polygon-ellipse [19] and polygonal approximation [20].

The total number of bits B required for storing or transmitting the contoured and encoded image is given as

$$B = \sum_{i=0}^P \{k_i [\lceil \log_2(x_{\max}^i - x_{\min}^i) \rceil + \lceil \log_2(y_{\max}^i - y_{\min}^i) \rceil] + \lceil \log_2(k_i) \rceil\} + R \times \lceil \log_2(R) \rceil + P \times [1 + \lceil \log_2(M) \rceil + \lceil \log_2(N) \rceil] \quad (5)$$

where R , P , and k_i denote the actual number of thresholds used for encoding, the total number of contours traced, and the number of control points in i th contour, respectively. The first term within the summation pertains to the bit requirement in transmitting the control points, and the second term corresponds to the number of control points. The remaining terms of (5) relate to the threshold indices, tag bits, and the corner points of the contour MBRs. Equation (5) does not include a few low-bit-rate terms that may have to be included in the file headers.

The PPI reflectivity fields generated from Houston, TX, NEXRAD Level II data collected on June 10, 2000 [34] are used in evaluating the performance of the method. The dimension of these reflectivity fields is 512×512 , and the bit depth is eight corresponding to a reflectivity interval of 0–64 dBZ. Standard NWS color table (legend) has been used to display the reflectivity distribution with a slight modification to display the background as white (the original background color is black). Four azimuth scans starting at 00:01:11, 00:06:11, 00:11:11, and 00:30:28 UTC on June 10, 2000 (designated as Field1, Field2, Field3, and Field4, respectively), all at an elevation angle of 0.35° , are used to generate the reflectivity PPI fields. In the rest of this section, it is assumed that the NWS thresholds are used to generate a contoured representation unless otherwise specified.

Before presenting the results, we comment on the computational feasibility of the scheme. The time required for encoding and decoding images using this scheme on a Pentium IV personal computer (1.8 GHz) is < 0.75 s. This is very small compared to the typical single-scan time of 20 s for WSR-88D radar. Hence, computational limitations are not likely to be a constraint for the adoption of this method in most situations.

Optimal choice of the parameters in the proposed method will result in better performance. Of particular importance are the number of thresholds, the smoothing length used in obtaining the smoothed contour, and the degree of B-spline interpolation. The effects of these parameters are studied with respect

TABLE II
COMPRESSION RATIO AS A FUNCTION OF PERCENTAGE
OF SMOOTHING LENGTH

Percentage	Field 1	Field 2	Field 3	Field 4
5	109.707	103.835	113.784	122.986
10	105.634	100.280	116.173	121.398
15	109.232	103.395	121.977	123.938
20	111.717	105.448	124.927	128.683

TABLE III
RMSE OF CONTOUR RECONSTRUCTION AS A FUNCTION
OF PERCENTAGE OF SMOOTHING LENGTH

Percentage	Field 1	Field 2	Field 3	Field 4
5	1.249390	1.164300	1.13833	1.52562
10	0.884273	0.884022	1.31600	1.18894
15	1.292630	1.113530	3.78364	3.60833
20	1.405210	1.539360	3.19848	3.53206

TABLE IV
RMSE OF CONTOUR RECONSTRUCTION AS A FUNCTION
OF DEGREE OF INTERPOLATION

Degree	Field 1	Field 2	Field 3	Field 4
1	0.785433	0.816937	1.21413	1.02699
2	0.884273	0.884022	1.31600	1.18894
3	0.993228	0.988289	1.49077	1.29947
4	1.115750	1.094810	1.51259	1.37056
5	1.212050	1.187960	1.62094	1.51071

to the achievable compression ratios, rmse of the retrieved contours, and visual fidelity of the reconstructed reflectivity fields.

1) *Effect of Smoothing Length*: Smoothing length is a key parameter that determines the extraction of control points. Specifying the smoothing length as a fraction of the contour length makes the smoothing process adaptive to the contour length. Very small smoothing lengths would result in little smoothing and improper location of control points. Large smoothing lengths would take the smoothed contour far away from the original contour and minimize the intersections between them, thus missing out many valid control points.

Tables II and III show the variation of the compression ratio and rmse as a function of the smoothing percentage, respectively. A spline of degree 2 is used (see Section IV-E2) to generate the results in Table III (Table II is independent of interpolation method). An optimal smoothing length should minimize the rmse while maximizing the compression ratio. The results show that this value is data dependent. A smoothing percentage of 10% is found to be near optimal for most data sets. The following sections assume that the control points are generated using a 10% smoothing length.

2) *Effect of Degree of Spline Interpolation*: The effectiveness of spline interpolation depends on the choice of the knot vector and the degree of spline interpolation. Since a uniform knot vector has been adopted in the proposed method, only the degree of spline interpolation is open to discussion. Table IV shows the variation of rmse with the degree of the spline.

The table suggests that rmse monotonically increases with the degree of interpolation. This is expected since the control points lie on the original contour itself, but the interpolated curve falls short of the points, with higher degrees of spline drawing the reconstructed contour farther away from the control points. This favors the use of low-degree splines for interpolation.

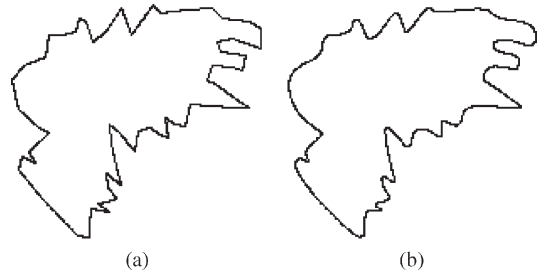


Fig. 4. Contour of Fig. 3 reconstructed with a spline of (a) degree 1 and (b) degree 2.

TABLE V
COMPRESSION RATIOS AS A FUNCTION OF THRESHOLDS

Field	NWS Thresholds	Augmented NWS Thresholds
1	105.634	59.0498
2	100.280	56.4905
3	116.173	63.7161
4	121.398	69.0125

Although a spline of degree 1 (which just joins successive control points using straight-line segments) yields marginally lower rmse than one of degree 2, it has a much more jagged or sawtooth-like appearance than the latter, as shown in Fig. 4. We therefore prefer the use of a degree 2 interpolation for the results in this paper. We subsequently show (Section V-A) that the performance of the degree 2 spline can be made to approach even closer to that of degree 1 by techniques such as control point stretching (which makes no sense for the degree 1 spline). However, in those applications which are sensitive to marginal rmse gains, the compression technique developed in this paper can well be used with a degree 1 spline interpolation.

3) *Effect of the Choice of Thresholds*: The choice of thresholds (both number as well as the values themselves) determines the visual fidelity of the proposed scheme. The NWS thresholds (Table I) are commonly used; however, in certain applications, these thresholds may be too coarse, particularly where the maximum reflectivity value is low. In such cases, the reconstructed image may not convey the same visual impact as that of the original reflectivity field, and significant meteorological information may be lost. To overcome this difficulty, a higher number of densely spaced thresholds may be chosen. We study here the effect of augmenting the NWS thresholds by introducing an extra level midway between each pair of successive levels.

Table V shows the variation of the compression ratio for the NWS and augmented NWS thresholds. As expected, the introduction of additional levels has degraded the compression ratio, as more contours would be generated, requiring more number of control points to be transmitted. However, this increase in the compressed data bandwidth is accompanied by enhancement in the quality of visual reproduction, as shown in Fig. 5.

F. Application to Other Fields

The contour-based compression scheme is readily applicable to other weather radar pictorial data fields as well. Besides reflectivity, the most common data field in the Doppler radar context is the distribution of radial velocities. Fig. 6 shows a sample radial velocity field and its retrieved form using the

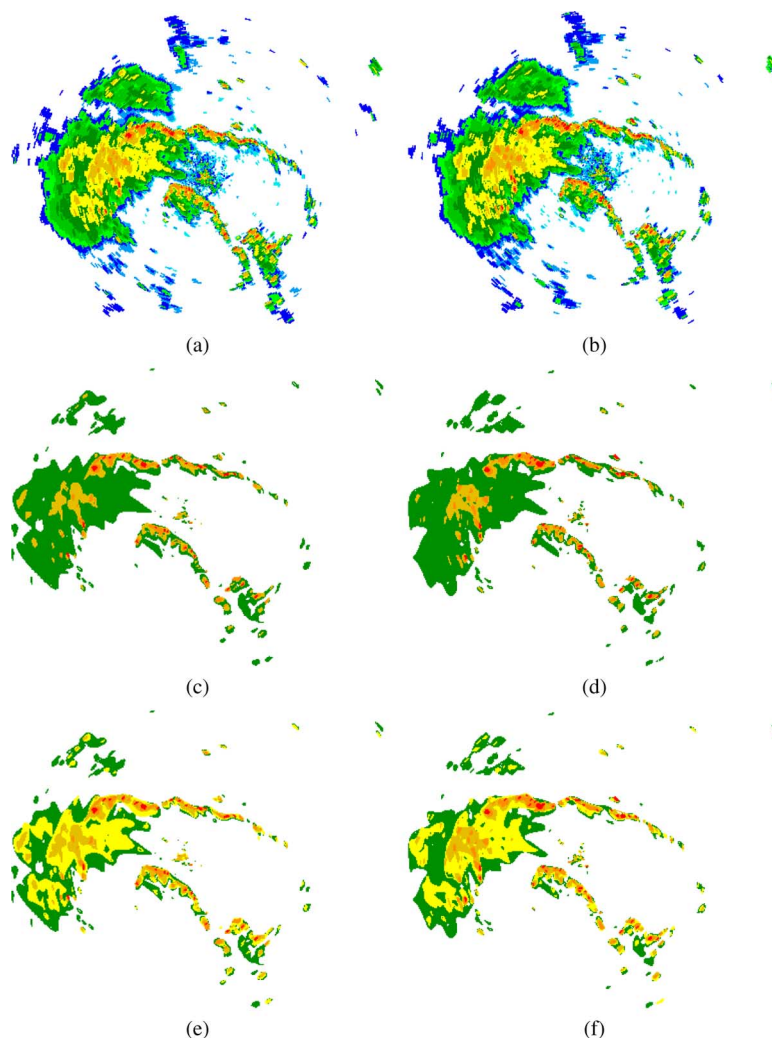


Fig. 5. Fidelity of reconstruction using the NWS and augmented NWS thresholds. (a) Original reflectivity PPI (Field 1). (b) Original reflectivity PPI (Field 2). (c) Retrieved Field 1 (NWS thresholds). (d) Retrieved Field 2 (NWS thresholds). (e) Retrieved Field 1 (augmented NWS thresholds). (f) Retrieved Field 2 (augmented NWS thresholds).

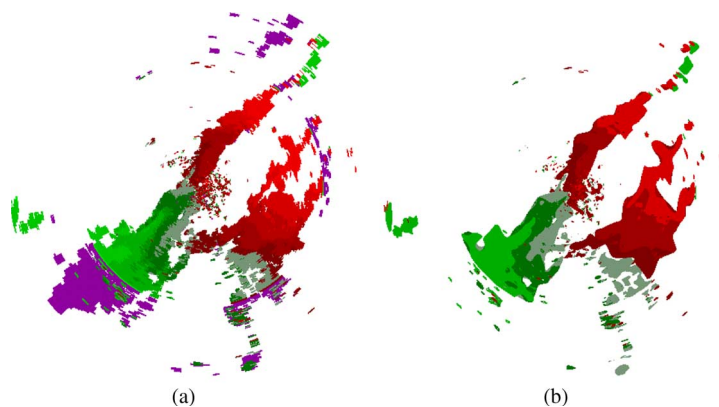


Fig. 6. Sample PPI velocity field from WSR-88D DWR in Level II format (elevation 0.48° , display 512×512 pixels with eight-bit depth). Radar location: Greer, SC, date, and time: January 23, 2006, 15:04:26 UTC [35]. (a) Original velocity PPI. (b) Retrieved field.

threshold values of -14 , -6 , -1 , 1 , 6 , and $14 \text{ m} \cdot \text{s}^{-1}$. The compression ratio in this case is ~ 40 .

The same method can also be applied to the other important Doppler moment, which is the spectrum width, in principle. However, in practice, it does not yield an efficient solution be-

cause of the usually noisy nature of such fields which results in a very large number of small contours. Further, the operational utility of such a procedure for higher moment fields is limited because it is normally only the reflectivity field which is viewed and interpreted by operators in the NWS-contoured form.

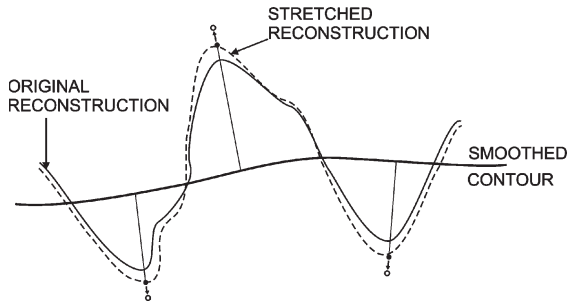


Fig. 7. Schematic showing principle of control point stretch.

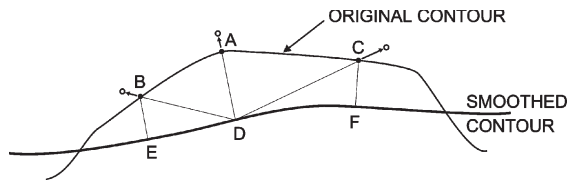


Fig. 8. Schematic showing the geometry of stretching the additional control points.

V. ENHANCEMENTS TO BASIC METHOD

A. Stretching of Control Points

Since the control points are chosen from among those lying on the contour itself, the reconstructed contour should ideally pass through them. However, as shown in Fig. 2, the spline-reconstructed contour will, in general, fall short of the control points because of its property of being completely confined within the convex hull formed by these points. This error may be offset, at least partially, by stretching the spread (distance) of the control points relative to the smoothed contour by a certain factor (called stretch factor; stretch % = {stretch factor - 1} × 100) so that the reconstructed contour moves closer to the original control points [36]. The exact direction and displacement of each control point are two critical parameters to be determined.

Fig. 7 shows a schematic where the solid dots denote the original control points and the spline curve bypasses them by a small distance. If these points were moved to the positions shown by the hollow dots, the reconstructed curve could move closer to the original control points. We perform this stretching by moving each extremal control point farther from the smoothed contour either by a fixed distance or, more realistically, by a distance proportional to its offset from the smoothed contour. The stretching of nonextremal control points (Section IV-B) follows a slightly different method in order to minimize the memory and/or computational requirements.

The method of stretching the control points is shown in Fig. 8. Here, point A denotes an extremal control point, while B and C denote two additional (nonextremal) control points. Let D be the point on the smoothed contour nearest to A. Point A is stretched in the direction \overrightarrow{DA} , with the displacement being a specified fraction of the length DA. By the same logic, B and C should be stretched in the directions \overrightarrow{EB} and \overrightarrow{FC} , respectively, where E and F denote their nearest points on the smoothed

TABLE VI
RMS CONTOUR RETRIEVAL ERROR AS A FUNCTION OF PERCENTAGE OF STRETCH

Stretch %	Field 1	Field 2	Field 3	Field 4
0	0.884273	0.884022	1.31600	1.18894
1	0.884273	0.884022	1.31600	1.18894
2	0.884273	0.884022	1.31600	1.18894
3	0.881816	0.883848	1.31313	1.18822
4	0.877065	0.882626	1.30654	1.17727
5	0.875231	0.882975	1.30622	1.18185
6	0.874312	0.879651	1.29831	1.17509
7	0.865074	0.875698	1.29073	1.17757
8	0.861537	0.881052	1.29611	1.16144
9	0.858454	0.878862	1.27919	1.11312
10	0.855078	0.888546	1.26470	1.11003
11	0.854420	0.892701	1.26477	1.11581
12	0.854326	0.894082	1.26503	1.11573
13	1.459560	0.888893	1.27045	1.13251
14	1.460060	0.890193	1.27197	1.12598
15	1.461980	0.888199	1.27183	1.12651

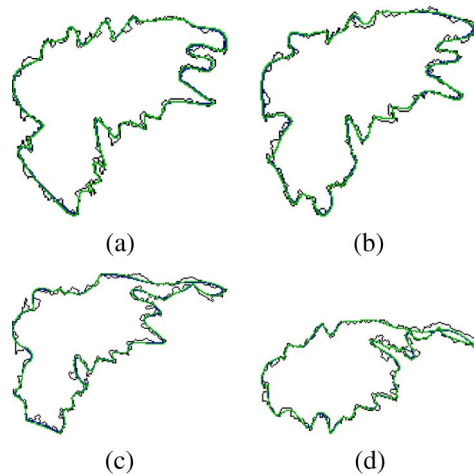


Fig. 9. Reconstructed contours for the (blue or dark gray) original and (green or light gray) stretched control points compared with (black) the original contour at four stages of evolution.

contour. However, considerable memory and/or computation is saved by stretching B and C along \overrightarrow{DB} and \overrightarrow{DC} , respectively, where point D is known *a priori*.

An optimum stretch factor would exist, since excessive stretching would overcompensate the miss distance and pull the retrieved contour too far away from the reference line, beyond the original control points. Table VI shows the variation of overall rmse (for all the contours in the indicated data field) with stretch factor. It may be concluded that the optimum stretch factor is in the range of 7%–12% for different data fields, while the corresponding improvement in rmse is between 1% and 7%. Fig. 9 shows the improvement in contour reconstruction obtained by stretching the control points of the contour of the largest area taken from each of the four data fields. If a uniform stretch factor were to be employed, a 10% stretch may be considered. However, since the stretching process is performed at the encoding end where sufficient computing resources may be available, an optimum stretch factor may be determined for each field or each contour independently rather than choosing a uniform value across all the fields.

TABLE VII
COMPRESSION RATIO AS A FUNCTION OF PERCENTAGE OF STRETCH

Stretch %	Field 1	Field 2	Field 3	Field 4
0	105.634	100.280	116.173	121.398
1	105.634	100.280	116.173	121.398
2	105.634	100.280	116.173	121.398
3	105.634	100.280	116.173	121.398
4	105.634	100.280	116.173	121.398
5	105.634	100.280	116.173	121.398
6	105.634	100.280	116.173	121.398
7	105.634	100.222	116.173	121.398
8	105.634	100.222	116.173	121.398
9	105.634	100.222	116.141	121.475
10	105.634	100.222	116.109	121.475
11	105.634	100.222	116.109	121.475
12	105.634	100.222	116.109	121.475
13	105.597	100.222	116.109	121.398
14	105.597	100.222	116.109	121.398
15	105.597	100.222	116.109	121.398

TABLE VIII
COMPARISON OF RMS CONTOUR RETRIEVAL ERRORS

Method	Contour 1 (66)	Contour 2 (69)	Contour 3 (84)	Contour 4 (66)
Spline	1.53863	1.60564	1.96551	1.97238
Polygonal	1.94767	1.83367	1.88074	1.77129
Polygon -Ellipse	2.11772	1.61304	1.51504	1.25601

It is worth mentioning here that the stretching process has a negligible effect on the compression ratio. The data load of the scheme primarily depends on the number of control points, which remains unchanged due to stretching. There would, however, be a weak effect on the data volume if any of the contour sizes is on the borderline such that a slight increase in their size would enhance the bit requirement of the MBR to the next level. This fact is corroborated by the occasional occurrence of small discrete jumps in the compression ratios in Table VII.

The performance of our spline reconstruction scheme incorporating stretching may be compared with the polygonal [20] and polygon-ellipse schemes [19]. Table VIII shows a comparison of the rmse obtained from the three methods for the contours shown in Fig. 9 using a common compression ratio, i.e., equal number of control points (indicated in parentheses). The results are mixed, suggesting that our scheme performs better than the other two schemes for only two of the four contours.

B. Bit Truncation for Enhancing Compression

The bit requirements for the control points depend on the size of the MBR for the contour. Truncating some of the least significant bits (LSBs) will further reduce the bit requirements of the control points, if the resulting loss of accuracy can be tolerated. Since truncation would be applied to all the points, significant compression gain may be achieved [36]. However, we retain, at least, the two most significant bits to preserve the contour shape. It should be noted here that the bit requirements for the x - and y -coordinates of the contour vary depending on the shape of the contour (elongated along x or y or both directions). Hence, a differential bit truncation method is adopted

TABLE IX
RMS CONTOUR RETRIEVAL ERROR AS A FUNCTION OF BITS TRUNCATED

No. of Bits	Field 1	Field 2	Field 3	Field 4
0	0.884273	0.884022	1.31600	1.18894
1	1.046950	1.015760	1.42590	1.27986
2	1.432900	1.402440	1.66883	1.58341
3	2.915660	2.404190	2.81206	2.61941

TABLE X
COMPRESSION RATIO AS A FUNCTION OF BITS TRUNCATED

No. of Bits	Field 1	Field 2	Field 3	Field 4
0	105.634	100.280	116.173	121.398
1	120.811	114.586	132.832	139.198
2	132.622	125.061	145.464	152.001
3	142.363	132.748	153.446	161.096

for the x - and y -coordinates. The numbers of bits truncated along the x - and y - directions n_{bx} and n_{by} are constant for all points on a given contour and are transmitted for each contour. It is worth pointing out here that truncation is applied to the control point coordinates referenced to the top left corners of their respective MBRs.

At the receiving end, the bits corresponding to the x - and y -coordinates of the control points are left-shifted by the number of LSBs removed at the encoding end. However, since truncation, in general, reduces the numbers representing both the coordinates, the reconstructed contour would be shifted toward the top-left corner with respect to the original. To correct for this shift, a bias compensation method has been adopted by adding $2^{n_{bx}-1}$ and $2^{n_{by}-1}$, respectively, to each x - and y -coordinate value, which correspond to the midvalues of the possible ranges of shift along each axis caused by the truncation.

It may appear that truncation may significantly affect the performance of the scheme by magnifying the stepped appearance of the contours. However, the powerful smoothing effect of spline interpolation effectively eliminates this problem for moderate levels of truncation. Experiments were conducted for 1-, 2-, and 3-bit truncation for various reflectivity fields, showing the rmse to increase by 8%–18%, 27%–62%, and 114%–230%, respectively, among the four data sets considered (Table IX). This percentage of increase may appear to be high, but the absolute increase is still acceptable being on the order of 1, 1.5, and 3 pixels for 1-, 2-, and 3-bit truncation, respectively. Table X shows the quantitative estimates of the tradeoff between the rmse and the compression ratio. In situations where the bandwidth is extremely limited, a 2-bit truncation (or even as much as 3 bits) may be opted for, resulting in savings of up to 25% (35% for 3 bits) in data volume while accepting an error of 1.5 pixels (3 pixels) in the contour reconstruction. Fig. 10 shows a comparison of the original and reconstructed contours for various levels of bit truncation.

Table XI shows the rmse obtained after compensating the bias induced due to bit truncation. The rms errors here for 1-, 2-, and 3-bit truncation are 8%–18%, 25%–57%, and 61%–140%, respectively, relative to the retrieved contour without any truncation. These errors are much lower than those for simple truncation, as shown in Table IX. Thus, the levels of compression suggested in Table X are now achievable with

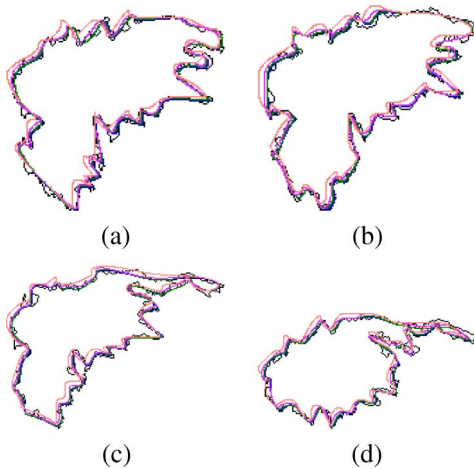


Fig. 10. Reconstructed contours for (blue or dark gray) 0-bit, (green or medium-dark gray) one-bit, (purple or medium-light gray) two-bit, and (pink or light gray) three-bit truncation compared with (black) the original contour at four stages of evolution.

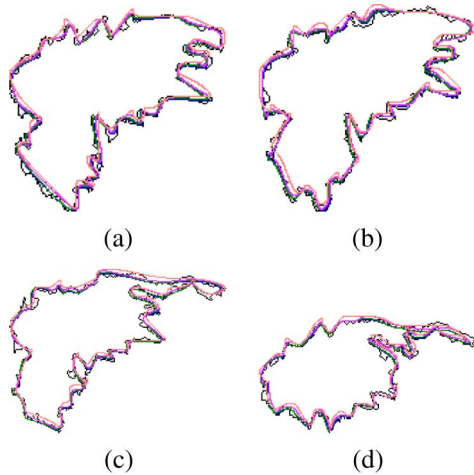


Fig. 11. Reconstructed contours for (blue or dark gray) 0-bit, (green or medium-dark gray) one-bit, (purple or medium-light gray) two-bit, and (pink or light gray) three-bit truncation after bias removal compared with (black) the original contour at four stages of evolution.

TABLE XI
RMS CONTOUR RETRIEVAL ERROR WITH TRUNCATED BITS AFTER BIAS REMOVAL

No. of Bits	Field 1	Field 2	Field 3	Field 4
0	0.884273	0.884022	1.31600	1.18894
1	1.046950	1.015760	1.42590	1.27986
2	1.389160	1.304700	1.64999	1.53904
3	2.119670	2.126820	2.12012	2.12088

only 1, 1.5, and 2 pixels of error for 1-, 2-, and 3-bit truncation. Fig. 11 shows the corresponding improvement in contour reconstruction.

VI. CONCLUSION

The very high volumes of data generated by weather radars require extreme levels of compression in certain circumstances. While such compression is bound to be lossy, the goal is to maximally preserve the meteorologically significant information in

the data fields. Since weather radar reflectivity data are most commonly utilized in a contoured form, this paper presents an efficient method in achieving very high levels of compression of such contoured data fields. It is shown that compression by more than two orders of magnitude is possible without visible degradation in the contour shapes. The method compares favorably with others devised for similar purposes. However, since all lossy compression schemes would entail some loss of information, the desirability and extent of compression should be carefully assessed with respect to the intended application.

The power of the compression method is derived from a control-point-based description which is able to capture the random undulations inherent in high-resolution weather radar contours. The method is also capable of scaling the number of control points and, consequently, the compression ratio, according to the need of retaining the undulations of various scales of fineness. A B-spline-based algorithm is used for faithful retrieval of the contours, and a second-degree spline is found to be the best choice.

The basic method can be enhanced and optimized in a variety of ways to improve its compression performance and fidelity. The continuity and smoothness of the data field can be adjusted by proper choice of the number of thresholds for contour generation, with attendant effects on compression ratios. The smoothing parameters used in generating the control points also provide a means of optimizing the quality of data retrieval. Errors introduced by the interpolation process can be minimized by a systematic stretching of the control points. A significant addition to the compression power of the method is possible by bit-manipulating the data stream in ways that minimize the loss of information.

Because of length constraints, we have been able to present detailed results only for limited data. However, we have tested the method on a wide range of data sets, including some very vigorous weather phenomena such as tornadoes associated with hurricane Katrina (New Orleans, LA and Mobile, AL, both on August 29, 2005) and the Oklahoma tornado on May 3, 1999, beside heavy and moderate precipitation activity over Houston, TX (June 10, 2000) and Greer, SC (January 23, 2006). The inferences of this paper are found to be valid for these diverse data.

REFERENCES

- [1] P. R. Mahapatra, *Aviation Weather Surveillance Systems: Advanced Radar and Surface Sensors for Flight Safety and Air Traffic Management*. London, U.K.: IEE Press, 1999, ch. 6.
- [2] P. Mahapatra and D. Zrnica, "Sensors and systems to enhance aviation safety against weather hazards," *Proc. IEEE*, vol. 79, no. 9, pp. 1234–1267, Sep. 1991.
- [3] R. J. Doviak and D. S. Zrnica, *Doppler Radar and Weather Observations*, 2nd ed. San Diego, CA: Academic, 1993.
- [4] P. Mahapatra and V. Makkapati, "Vector reconstruction and mosaicing from multiple Doppler weather radar velocity data," in *Proc. Int. Geosci. Remote Sens. Symp.*, Anchorage, AK, Sep. 2004, vol. 6, pp. 3868–3871.
- [5] A. Jain, "Image data compression: A review," *Proc. IEEE*, vol. 69, no. 3, pp. 349–388, Mar. 1981.
- [6] A. N. Netravali and J. O. Limb, "Picture coding: A review," *Proc. IEEE*, vol. 68, no. 3, pp. 366–406, Mar. 1980.
- [7] M. J. Weinberger and G. Seroussi, "Lossless compression of continuous-tone images," *Proc. IEEE*, vol. 88, no. 11, pp. 1797–1809, Nov. 2000.

- [8] N. Memon and K. Sayood, "Lossless image compression: A comparative study," in *Proc. SPIE—Still-Image Compression*, Mar. 1995, vol. 2418, pp. 8–20.
- [9] A. Kruger and W. Krajewski, "Efficient storage of weather radar data," *Softw.—Pract. Exp.*, vol. 27, no. 6, pp. 623–635, Jun. 1997.
- [10] V. Lakshmanan, "Lossless coding and compression of radar reflectivity data," in *Proc. 30th Int. Conf. Radar Meteorol.*, Jul. 2001. CDROM. [Online]. Available: <http://ams.confex.com/ams/30radar/30radar/abstracts/21592.htm>
- [11] O. Egger, P. Fleury, T. Ebrahimi, and M. Kunt, "High-performance compression of visual information—A tutorial review—Part I: Still pictures," *Proc. IEEE*, vol. 87, no. 6, pp. 976–1011, Jun. 1999.
- [12] M. Reid, R. Millar, and N. Black, "Second-generation image coding: An overview," *ACM Comput. Surv.*, vol. 29, no. 1, pp. 3–29, Mar. 1997.
- [13] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, Apr. 1991.
- [14] C. Christopoulos, A. Skodras, and T. Ebrahimi, "The JPEG2000 still image coding system: An overview," *IEEE Trans. Consum. Electron.*, vol. 46, no. 4, pp. 1103–1127, Nov. 2000.
- [15] R. J. Campbell and P. J. Flynn, "A survey of free-form object representation and recognition techniques," *Comput. Vis. Image Underst.*, vol. 81, no. 2, pp. 166–210, Feb. 2001.
- [16] H. Freeman, "On encoding arbitrary geometric configurations," *IRE Trans. Electron. Comput.*, vol. 10, pp. 260–268, 1961.
- [17] D. H. Douglas and T. K. Peucker, "Algorithms for the reduction of the number of points required to represent a line or its caricature," *Can. Cartogr.*, vol. 10, no. 2, pp. 112–122, 1973.
- [18] C. Teh and R. Chin, "On the detection of dominant points on digital curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 8, pp. 859–872, Aug. 1989.
- [19] J. L. Gertz and R. D. Grappel, "Storage and transmission of compressed weather maps and the like," U.S. Patent 5 363 107, U.S. Pat. Off., Washington, DC, Nov. 8, 1994.
- [20] D. Burdon, "System and method for the adaptive mapping of matrix data to set of polygons," U.S. Patent 6 614 425, U.S. Pat. Off., Washington, DC, Sep. 2, 2003.
- [21] E. R. Hilgendorf and R. H. Johnson, "A study of the evolution of mesoscale convective systems using wsr-88d data," *Weather Forecast.*, vol. 13, no. 2, pp. 437–452, 1998.
- [22] M. I. Skolnik, *Introduction to Radar Systems*, 3rd ed. New York: McGraw-Hill, 2000.
- [23] C. D. Boor, *Practical Guide to Splines*. New York: Springer-Verlag, 1978.
- [24] P. Mahapatra and V. Makkapati, "Ultra high compression for weather radar reflectivity data storage and transmission," in *Proc. 21st Int. Conf. Interactive Inf. Process. Syst. Meteorol., Oceanogr., Hydrol.*, Jan. 2005. CDROM. [Online]. Available: http://ams.confex.com/ams/Annual2005/techprogram/paper_82973.htm
- [25] M. Alder, *An Introduction to Pattern Recognition*. Osborne Park, Western Australia: HeavenforBooks.com, 1997, ch. 2.
- [26] T. Pavlidis, *Algorithms for Graphics and Image Processing*. Berlin, Germany: Springer-Verlag, 1982, ch. 7.5.
- [27] F. Attneave, "Some informational aspects of visual perception," *Psychol. Rev.*, vol. 61, no. 3, pp. 183–193, 1954.
- [28] S. Biswas, "Contour coding through stretching of discrete circular arcs by affine transformation," *Pattern Recognit.*, vol. 34, no. 1, pp. 63–77, 2001.
- [29] D. DeMenthon, V. Kobla, and D. Doermann, "Video summarization by curve simplification," in *Proc. 6th Int. Conf. Multimedia*, 1998, pp. 211–218.
- [30] J. D. Foley, A. V. Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice in C*, 2nd ed. Reading, MA: Addison-Wesley, 1995.
- [31] S. Grgic, M. Grgic, and M. Mrak, "Reliability of objective picture quality measures," *J. Electr. Eng.*, vol. 55, no. 1/2, pp. 3–10, 2004.
- [32] H. Sheikh and A. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.
- [33] P. L. Rosin, "Techniques for assessing polygonal approximations of curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 6, pp. 659–666, Jun. 1997.
- [34] D. Priegnitz, *Interactive Radar Analysis System*. [Online]. Available: <http://www.ncdc.noaa.gov/oa/radar/iras.html>
- [35] *Integrated Radar Data Services*. [Online]. Available: <https://www.radarservices.org/>
- [36] P. Mahapatra and V. Makkapati, "Performance enhancement of a spline-based method for extreme compression of weather radar reflectivity data," in *Proc. ISSPIT*, Dec. 2005, pp. 147–152.



Vishnu V. Makkapati (M'07) received the B.E. (Honors) degree in electrical and electronics and the M.Sc. (Honors) degree in mathematics from the Birla Institute of Technology and Science, Pilani, India, in 2000, and the M.Sc. (Engg.) degree from the Indian Institute of Science, Bangalore, in 2007.

He was with the IBM India Software Laboratory until April 2001 and, then, with the Honeywell Technology Solutions Laboratory, India, until December 2006, reaching the level of Principal Engineer and receiving numerous awards and citations for his innovative work. He is currently a Senior Scientist with Philips Research Asia-Bangalore, Philips Electronics India Ltd., Bangalore. His research interests include image processing, computer vision, image compression, and wavelet transform.

Mr. Makkapati was the recipient of a National Merit Scholarship and a State Merit Scholarship during his studies.



Pravas R. Mahapatra received the B.Sc.(Engg.) degree with Honors from Regional Engineering College, Rourkela, Orissa, India, and the Master of Engineering and Ph.D. degrees from Indian Institute of Science, Bangalore.

He has been with the faculty of the Department of Aerospace Engineering, Indian Institute of Science, Bangalore, since 1970, where he currently holds the Prof. Satish Dhawan Chair Professorship. His teaching and research interests cover a broad range of topics in the field of aerospace and electronic systems. He has been a Senior Research Associate with the U.S. National Research Council, working at NOAA National Severe Storms Laboratory and NASA/Caltech Jet Propulsion Laboratory. He has also been a visiting faculty member at the University of Oklahoma and the University of Rome at Tor Vergata, Italy. He has published extensively in international journals and conferences. He has authored the first ever book on aviation weather surveillance systems, which was published by AIAA and IEE.

Prof. Mahapatra is a Fellow of the Indian National Academy of Engineering, Fellow of the Institute of Electronics Telecommunication Engineers, Professional Member of the U.S. Institute of Navigation, and Life Member of the Aeronautical Society of India, serving as its Vice President for five years. He received the 1993 IEEE Donald G. Fink Prize for his work on aviation weather surveillance systems. He was also the recipient of the Academic Excellence Award from the Prime Minister of India and the Astronautical Society of India Award for Space Science and Application from the President of India, both in 2002.