19.10   ULTRA HIGH COMPRESSION FOR WEATHER RADAR REFLECTIVITY DATA STORAGE AND TRANSMISSION

Pravas R. Mahapatra *
Indian Institute of Science
Bangalore, India

Vishnu V. Makkapati
Honeywell Technology Solutions Lab
Bangalore, India

ABSTRACT

Bandwidth is a premium resource in ground-air and air-ground communication links. Transmission of raw Doppler weather radar data over these links incurs significant latency and high cost. Traditional image compression techniques utilizing spatial redundancy of pictorial data for reducing the bit-rate achieve at most an order-of-magnitude of compression ratios. An efficient compression scheme for reflectivity data based on contour representation of weather phenomena is presented in this paper. The method, incorporating automatic contouring, control point generation, spline-based reconstruction, and contour filling, offers compression ratios in excess of a hundred while preserving meteorologically useful features.

1.   INTRODUCTION

Weather is a common cause for flight delays and a contributing factor for aviation accidents. In recent years Doppler Weather Radar (DWR) Skolnik (2000), Doviak and Zrnic (l993) has emerged as a potent tool for obtaining accurate and high-resolution weather data which can be used by pilots and meteorologists for hazard assessment and flight planning. Extensive 3-D weather data are generated by numerous ground-based radars such as those in the US WSR-88D network. To be useful to the pilot, these data must be transmitted to the cockpit via air-ground communication links. Most large aircraft also carry weather radars on board which provide a view of the local weather at the flight altitude. These data are often transmitted to the ground via air-ground links for viewing by meteorologists and also for integration with ground radar and other meteorological data.

Bandwidth is a premium resource in data transmission between ground and aircraft. Since DWR data is inherently voluminous, its transmission over

---
Corresponding author address: Pravas R. Mahapatra, Department of Aerospace Engineering, Indian Institute of Science, Bangalore - 560 076, India; e-mail: pravas@aero.iisc.ernet.in.
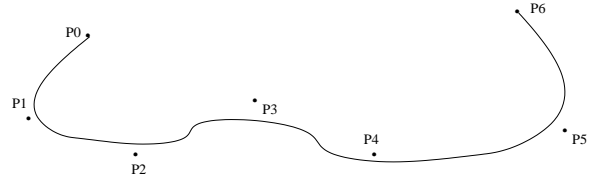
Figure 1: B-Spline representation of a curve

such links is very expensive. It also generally involves significant data latency when transmitted over commonly available links. It is therefore necessary to compress radar-generated weather pictures as far as possible for transmission to and from aircraft. Compression of pictorial data is a subject of much study, and many different types of algorithms are available for the purpose Egger et al. (1999). However, general-purpose picture compression algorithms seldom achieve compression ratios better than an order of magnitude. Contour-based picture representation methods Barr (1981) have the potential to achieve higher orders of compression. Attempts have been made Gertz and Grappel (1994), Burdon (2003) to represent weather pictures in terms of elliptical or polygonal contours which may then be represented in terms of a few parameters. However, such shapes imposes severe constraints on the types of pictures that can be faithfully modeled. This paper reports a method of compressing generalized weather radar reflectivity data by factors exceeding a hundred, while preserving their meteorologically significant features.

2.   SPLINE MODELING OF CLOSED CURVES

The curve defined by a parameter t:

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

is a B-spline Boor (1978). $C(t)$ denotes the point on the curve at the parametric value t. The curve is drawn for various values of t, ranging from t_{min} to t_{max}, typically having values 0 and 1 respectively.

Table 1: NWS Reflectivity Thresholds

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

The shape of the B-spline curve can be controlled by $(n + 1)$ control points, denoted by P_i, which are points in the object space. The positions of the control points are chosen so as to achieve the shape of the curve being modeled. The B-spline curve does not necessarily pass through the control points, but always lies within the convex hull of these points (Fig. 1).

A 'knot vector' is defined as

$$T = \{t_0, t_1, ..., t_m\} \quad (2)$$

where the constants $t_i \in [0, 1]$, called 'knot values', are strictly in nondecreasing order. Define the degree p of the curve as

$$p \equiv m - n - 1 \quad (3)$$

The curve becomes smoother as the degree of the curve increases. The knot values $t_{p+1}, ..., t_{m-p-1}$ are called internal knots. A uniform B-spline curve is one with equally spaced internal knots.

The basis function $N_{i,p}(t)$ decides the extent to which a control point controls the curve at a parametric value t.

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

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

$$+ \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t) \quad (4)$$

These functions have a non-zero 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 1. However, the sum of basis function values is not equal to 1 in the intervals $[t_0, t_p)$ and $(t_{n+1}, t_{n+p+1}]$ and no affine combination of the control points is possible. The curve is thus defined only in the interval $[t_p, t_{n+1}]$, and end point interpolation is not possible. End point interpolation is achieved by considering the first $p + 1$ and last $p + 1$ knots 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.

B-splines provide a common mathematical representation of standard analytic shapes as well as free-form curves, and are projection invariant. Such a representation is chosen here for the following additional advantages:

1. Changes to a control point only affects 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 towards that position

4. Closed curves can be represented by making the first and last points the same.

3. ALGORITHM STEPS

3.1 Contour Tracing

The 2-D reflectivity distribution is converted into a binary image by applying a threshold. It is convenient to choose one of the threshold values specified by the National Weather Service (NWS) (Table 1) corresponding to different levels of severe weather phenomena Mahapatra (1999). The boundary separating an area of 0's from an area of 1's is called a contour. Contours enclosing a zone of 1's are called 'region contours', and boundaries within these region contours which have only 0's within are called 'holes'. It is possible for region contours and holes to be nested, i.e. there may be region contours within holes, and so on.

To trace the contours of the binary images, we use the radial sweep algorithm Alder (1997). This algorithm does not automatically trace internal structures of contours such as holes, region contours inside them, and nested contours. Further, it traces only one contour at a time, only after the contour has been initiated through a search procedure. To overcome these limitations, we use the scheme proposed in Pavlidis (1982), applying it recursively to trace nested holes and region contours. While tracing a contour, the scheme increments the pixels on the contour by 1, i.e. assigns a value of 2. On overlapping contour segments (e.g. a hole occurring tangentially within a region) the pixels are assigned a value of 3. The direction of interior of a
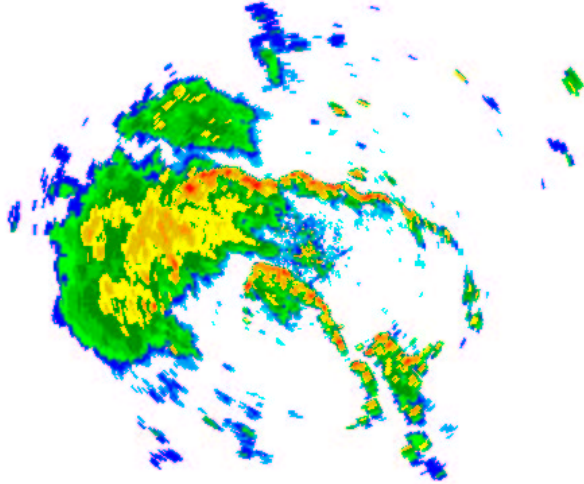
Figure 2: Original PPI image

contour is determined using chain codes. The pattern of three successive pixels is utilized for tracing holes as well as region contours.

3.2 *Extracting Control Points*

The aim here is to capture the shape of the curve with a limited number of significant points. A novel scheme for extracting the control points is proposed in this paper. This scheme averages the given contour over a certain neighborhood of each point to obtain a smoothed curve. The deviation of each point on the original contour from the nearest point on the smoothed curve is calculated. To start with, the scheme determines a point on the original contour which has a deviation ≤ 1 pixel from the nearest point on the smoothed curve and consider it as a zero-crossing point. All zero-crossing points on the contour are determined by traversing around the full contour.

Each point on the original contour that lies between two consecutive zero-crossings and has a maximum distance from the smoothed curve is considered as a control point. However, if the length of the contour between any two zero crossings exceeds a certain threshold, an extra control point is added between the control point as determined above and the zero-crossing points on either side of it. This scheme may not extract proper control points for smooth contours and small contours (e.g. contours with < 20 points). The control points for such contours are extracted by considering uniformly spaced points along the contour.

3.3 *Contour Transmission and Reconstruction*

The thresholds used for encoding are stored in a look up table (LUT) and only the indices need to be transmitted to minimize bit requirements. Since there may be multiple closed contours for any given threshold, each closed contour is numbered before transmission. To distinguish region contours from hole contours, a tag bit of 0 and 1 respectively may be used.

The coordinates of the control points for each contour are transmitted to the receiving end. To minimize data volume, the coordinates of the control points for each contour are referenced with respect to the top left corner of the minimum bounding rectangle (MBR) for that contour. The coordinates (x_{min_i}, y_{min_i}) of this corner point needs to be transmitted once per contour.

At the receiving end the contour is reconstructed by using B-spline interpolation. The control points are restored to being referenced with respect to the original picture coordinates by adding (x_{min_i}, y_{min_i}) to their transmitted coordinates. Let $(n_i + 1)$ be the number of control points received. The parameter t is incremented uniformly in the range [0, 1] in steps of $\frac{1.0}{(n_i+1) \times k}$, where k is an integer. Higher values of k may lead to duplicate points which may be discarded; any gaps resulting from this are filled by using Bresenham's line drawing algorithm Foley et al. (1995).

The interiors of reconstructed region contours should be filled with the threshold value of the contour using a region-filling algorithm. The regions corresponding to higher thresholds are in general smaller than those for the lower thresholds while the holes for the former are larger than those for the latter thresholds. The filling scheme should fill these regions efficiently without any redrawing. In the image reconstruction algorithm, the contours of the regions as well as the holes corresponding to a given threshold are retrieved on the basis of the control points using the B-spline fit. Regions and holes are identified by means of the tag bit transmitted. The area within each region contour but outside the holes is filled with the color value corresponding to the given threshold using the boundary fill algorithm Foley et al. (1995). An arbitrary pixel inside the region contours but outside any holes is located Pavlidis (1982) and is used as the seed for the boundary fill process. This sequence is repeated for all the thresholds used for contouring the image, with higher-threshold sections being successively superposed on those of lower threshold values. This yields a composite contoured image of
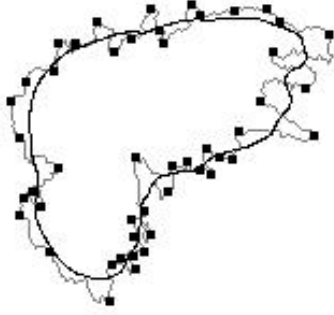
Figure 3: A raw contour (lighter line) obtained by 30-dB thresholding, and its smoothed version (darker line). The small black squares are the control points.

Figure 4: The raw contour of Fig. 3 (black line), and its reconstruction using our scheme (red), the method by Burdon (2003) (green) and the method by Gertz and Grappel (1994) (blue).

the reflectivity field to the pilot.

4. PERFORMANCE EVALUATION

The performance of the compression scheme is evaluated by the compression ratio it provides. The total number of bits B required for transmitting the contoured and encoded image is given as

$$B = \sum_{i=0}^{P} \{k_i \left(log_2(m_i) + log_2(n_i)\right) + log_2(k_i)\}$$
$$+ R \times log_2(R) + P \times (1 + log_2(M) + log_2(N)) \quad (5)$$

where M and N are the width and height of the picture, and 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 for transmitting the control points (referenced to the top left of the MBR, with width m_i and height n_i), which constitutes the dominant data load for transmission, 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 point 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 compression ratio (CR) can then be expressed as

$$CR = \frac{M \times N \times log_2(I)}{B} \quad (6)$$

where I denotes the peak reflectivity value in the given image.

The compression algorithm has been tested using the Plan Position Indicator (PPI) images of radar reflectivity Level II data from a WSR-88D radar. The size of the image considered is 512 × 512 pixels. The reflectivity values are represented with 8-bit resolution, yielding a scale of 0-255 corresponding to a maximum reflectivity interval of 0-64 dBZ. The NOAA standard color table ('legend') is used for displaying the reflectivity distribution, with a slight modification to display the background as white (the original background color is black) (Fig. 2).

The largest-area contour appearing in the 30-dB-threshold section of the picture is shown in Fig. 3 in raw form and after smoothing it over 10% of the contour length. Following the procedure outlined in Sec. 3.2 we obtain 66 control points. A second degree spline fitted with these 66 control points is shown in Fig. 4. The Root Mean Square (RMS) departure of the reconstructed contour from the original is found to be 1.1197. For comparison, the contours obtained by the methods outlined in Gertz and Grappel (1994) and Burdon (2003) are also included in Fig. 4. These contours yield RMS departures of 1.1348 and 1.2104 respectively which are higher than our value.

The compression ratio for the scheme as evaluated from equation (6) is 112. This level of compression is very high compared with any of the general-purpose image compression methods. The reconstructed image (Fig. 5) is found to retain all of the meteorologically significant features of the original image.

The time required for encoding and decoding images using this scheme on a Pentium 4 PC (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 this scheme can be used for real time or quasi-real time compression of reflectivity data.
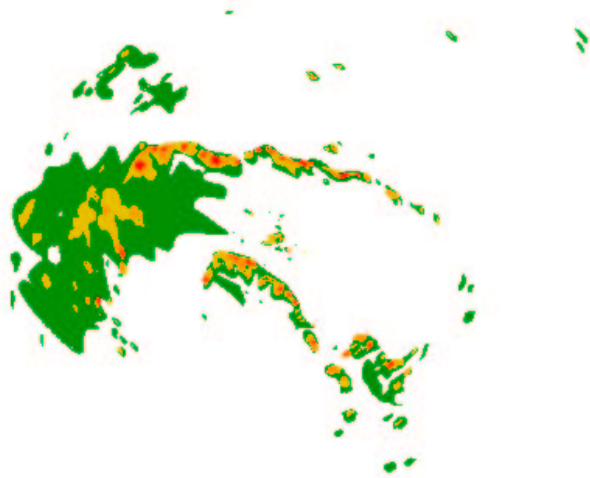
Figure 5: Reconstructed PPI image

5. CONCLUSIONS

This paper presents a scheme for very high compression of weather radar data for transmission between ground and aircraft, for which bandwidth is a scarce resource. Compared to the single-order-of-magnitude compression ratios possible with general-purpose picture compression algorithms, the present algorithm yields ratios of over a hundred while substantially preserving the meteorological information. Our method is also found to be superior to existing contour based methods.

Higher levels of compression than what is demonstrated here can be achieved by reducing the number of control points transmitted, but a drastic reduction can result in inaccurate representation of the contour. A lossless contour encoding scheme can be employed in those cases which require exact reproduction of the contour(s).

REFERENCES

Alder, M., 1997: *An Introduction to Pattern Recognition*, HeavenforBooks.com, Osborne Park, Western Australia, chapter 2.

Barr, A. H., 1981: Superquadrics and angle-preserving transformations. *IEEE Computer Graphics and Applications*, **1**, 11 – 23.

Boor, C. D., 1978: *Practical Guide to Splines*. Springer Verlag, New York, NY.

Burdon, D., 2003: System and method for the adaptive mapping of matrix data to set of polygons. *U.S. Patent 6614425*.

Doviak, R. J. and D. S. Zrnic, l993: *Doppler Radar and Weather Observations*. Academic Press, San Diego, CA, second edition.

Egger, O., P. Fleury, T. Ebrahimi, and M. Kunt, 1999: High-performance compression of visual information - A tutorial review - Part I: Still pictures. *Proc. of the IEEE*, volume 87, 976 – 1011.

Foley, J. D., A. V. Dam, S. K. Feiner, and J. F. Hughes, 1995: *Computer Graphics: Principles and Practice in C*. Addison-Wesley, Reading, MA, second edition.

Gertz, J. L. and R. D. Grappel, 1994: Storage and transmission of compressed weather maps and the like. *U.S. Patent 5363107*.

Mahapatra, P. R., 1999: *Aviation Weather Surveillance Systems: Advanced Radar and Surface Sensors for Flight Safety and Air Traffic Management*, IEE Press, London, UK, chapter 6.

Pavlidis, T., 1982: *Algorithms for Graphics and Image Processing*, Springer Verlag, Berlin, Germany, chapter 7.5.

Skolnik, M. I., 2000: *Introduction to Radar Systems*. McGraw-Hill, New York, NY, third edition.