DEM Compression Based on Integer Wavelet Transform

CHEN Renxi  LI Xinhui

Abstract  DEM data is an important component of spatial database in GIS. The data volume is so huge that compression is necessary. Wavelet transform has many advantages and has become a trend in data compression. Considering the simplicity and high efficiency of the compression system, integer wavelet transform is applied to DEM and a simple coding algorithm with high efficiency is introduced. Experiments on a variety of DEM are carried out and some useful rules are presented at the end of this paper.

Keywords  DEM; wavelet transform; data compression

CLC number  P208

Introduction

Digital elevation model (DEM) data is so large that minimizing storage space is necessary. There are several compression methods for DEM: ① converting regular gridded DEM to TIN DEM; ② using entropy encoding methods directly; ③ using generally available compressing algorithms for images[1-4].

In recent years, wavelet transform has obtained great success in the field of image compression because wavelets have more advantages than traditional compressing methods. Especially, integer wavelet transform (IWT), which is based on second-generation wavelets, lift scheme, has the advantages of simplicity and low computational complexity. Based on integer wavelets, we carried out some investigations and experiments for DEM data compression by using a simple and highly efficient coding algorithm.

1  Differential coding

Discrete wavelet transform has been successfully used in the field of image compression, and some sophisticated algorithms have been presented, such as EZW[5], SPIHT[6]. These algorithms use tree structure and perform excellently, but require the image size to be 2^n (or make special processing if not be 2^n). On the other hand, these algorithms have fairly high computational complexity and require mass memory storage. In this paper, instead of using these algorithms directly, we present a simple coding algorithm based on a differential coding method. Our method, which we describe in this paper, is inspired by WDR algorithm in Reference [7].

The framework of the coding method is shown in Fig.1. Firstly, an initial threshold value T is chosen, and then significant coefficients scan and refinement process are performed in loops. After each loop, the threshold T is reduced in half. Obviously, the algorithm is a method based on bit-plane encoding. In
each loop, if the absolution of the wavelet coefficient is greater than threshold $T$, then it is called a significant coefficient, otherwise it is called insignificant.

## 2 IWT and encoding

The brief workflow of the method can be described as follows. The data redundancy is removed by discrete wavelet transform, and then the wavelet coefficients are quantified. Then, entropy-coding methods (such as the Huffman coding or arithmetic coding) are used to generate compressed bit-steam. Fig.2 shows the whole framework.

![Framework of DEM compression](image)

### Fig.2 Framework of DEM compression

### 2.1 Preprocessing

DEM data is often stored as float number and cannot use IWT directly. In order to use integer wavelet transform, preprocessing should be performed prior to the transform, including elevation unit conversion and elevation displacement.

1) Elevation unit conversion. Since integer transform is used, the computational precision is one unit, and this cannot meet the accuracy requirement of the reconstructed data. For example, an elevation value $H_i = 235$ m in loss compression case, the reconstructed value is not equal to the original value and the error is at least 1 m. That is to say, the reconstructed value may be 234 m or 236 m. If the error found is within 0.5 m, then IWT cannot meet the accuracy requirements. A simple solution to this problem is to convert “m” unit to “dm” or “cm” unit, then the precision can reach 0.1 m or 0.01 m.

2) Elevation displacement. The medium elevation value $V$ of the whole DEM is computed and then subtracted from each point. Thus, the absolution of elevation value becomes smaller, which helps to reduce the overhead in programs and improve the compression ratio.

### 2.2 Integer wavelet transform

After preprocessing, all the original data become integers and are then transformed into 2D wavelet space using IWT. In comparison with traditional wavelet transform, IWT has more advantages: ① IWT uses lift scheme, and calculating speed is improved; ② IWT is absolutely reversible (data can be reconstructed without any error), which makes both loss and lossless compression possible.

In data compression, high compression ratio (CR) is always incompatible with low complexity. In order to look after both sides, we should make a middle course. In Reference [8], Adams listed 12 commonly used integer wavelet bases, each of which has different computational complexity and compressing performance. Table 1 shows the total operation times of each wavelet base (numbers in brackets denote operation times in inverse transform).

| Wavelet       | Add | Shift | Multiply | Total |
|---------------|-----|-------|----------|-------|
| 5/3           | 5   | 2     | 0        | 7     |
| 2/6           | 5   | 2     | 0        | 7     |
| SPB           | 7(8)| 4(3)  | 1(0)     | 12(11)|
| 9/7-M         | 8(9)| 2(3)  | 1(0)     | 11(12)|
| 2/10          | 7(10)|2(6)  | 2(0)     | 11(16)|
| 5/11-C        | 10  | 3     | 0        | 13    |
| 5/11-A        | 10  | 3     | 0        | 13    |
| 6/14          | 10(11)|3(5) | 1(0)     | 14(16)|
| SPC           | 8(10)|4(5)  | 2(0)     | 14(15)|
| 13/7-T        | 10(12)|2(4) | 2(0)     | 14(16)|
| 13/7-C        | 10(12)|2(4) | 2(0)     | 14(16)|
| 9/7-F         | 12(26)|4(18)| 4(0)     | 20(44)|

Comparing the computational complexity of all the bases listed in Reference [8], 5/3 and 2/6 wavelets are the lowest, and 9/7-F is the highest. If the processing speed is important (in real-time computational case), then 5/3 wavelet is the first candidate. Although 2/6 wavelet has the same computational complexity as 5/3, it is not good as the latter in compressing performance.

Experiments in Reference [8] demonstrate that: ① in lossless compression case, the performance difference between the best and the worst is no more than 2%, but 5/3 wavelet is the best for images with many details; ② in loss compression case, 9/7-F wavelet is the best at low bit rate ($\leq 0.5$ bit/pixel), but when the bit rate is heightened, 5/3 wavelet obtains better performance.

In low bit rate loss compression case, 9/7 wavelet
is the best. But for DEM compression, high fidelity is required so the bit rate should not be too low. Sometimes, lossless or near lossless compression is required for DEM. On the other hand, computational speed is very important for mass data processing. Thus, 5/3 wavelet is very suitable for DEM data compression.

Unlike traditional wavelet transform, 5/3 IWT is not an orthogonal transform. The coefficients in IWT do not have the same distribution characteristic as traditional wavelets, so bit plane scanning encoding method cannot be used directly. In order to use bit plane scanning encoding, coefficients in all subbands should be multiplied by a weighting factor. For example, 3-levels wavelet transform weighting coefficients for each sub-band are shown in Fig.3.

![Weighting coefficients for 3-levels 5/3 IWT](image)

### 2.3 Quantification and encoding

According to the theory of wavelet transform, data volumes after a transform are the same as before the transform. But the power is more compact in order to achieve easier compression. In general, coefficients should be quantified before encoding. In this paper, the quantifying method is from the idea of SPIHT. For each coefficient, the highest bit is the most significant. A simple quantifying scheme is to remove $n$ bits from the lowest to the highest, that is to say, quantified value is derived by right-shifting every coefficient by $n$ bits, $$x_{q_i} = \text{sgn}(x_i) |x_i| / 2^n.$$ Although this is equivalent to scalar quantification it is comparatively better. The shift bit number $n$ is inputted by the user. After quantification, every coefficient can be encoded by using the bit-plane scanning method (same as SPIHT). We use arithmetic coding as the entropy encoder since it performs better than the Huffman encoding. Unlike SPIHT algorithm, the coder presented in this paper encodes each subband separately so that the bit-stream has a progressive resolution transmission characteristic. This is very useful for mass DEM previewing, data retrieval and multi-resolution expression.

### 3 Experiments

According to the algorithm mentioned above, we made C++ program and carried out some experiments. In general, the terrain varies greatly in dissimilar areas and this affects the compressing effect. Therefore, the compression ratio is greatly affected by the complexity of the terrain, the range of the elevation and the resolution of the DEM. In the experiments, four dissimilar types of DEM data with the same spatial resolution (25 m) are chosen as shown in Fig.4. For image compression, 5-6 levels wavelet transform is often used. In our experiments, we found that 5 levels wavelet transform is appropriate.

![DEM data of different areas](image)

For DEM data compression, the error requirement is relevant to the scale and the intention of the application, and there is not an appropriate evaluation criterion. In order to preserve high fidelity, the maximal error is limited within 1.0 m, and it is almost near lossless compression. In order to quantitatively express the complexity of the terrain, fractal dimension is used which is computed by fractional Brownian
motion model. The larger the fractal dimension is, the more complicated the terrain is. Each point in the original data is stored as float number (4 bytes per point). Table 2 shows the experiment results.

Table 2  Experiment on different DEM data

| No. | Range /m | Fractal dimension | RMSE /m | Max error/m | CR |
|-----|----------|-------------------|---------|-------------|----|
| (a) | 20-280   | 2.10              | 0.12    | 0.5         | 11.95 |
| (b) | 102-610  | 2.25              | 0.14    | 0.7         | 6.51  |
| (c) | 50-350   | 2.32              | 0.12    | 0.5         | 6.32  |
| (d) | 1,456-1,958 | 2.56             | 0.13    | 0.6         | 4.70  |

It is obvious that the compression ratio varies greatly for different DEM data. In simple terrain areas (such as plain, foothill and large scale mountain areas), the compression ratio is high; but in complex areas (such as little scale mountain or fragmental terrain areas), the compression ratio is low.

For the terrain with the same complexity, relative height will also affect the compression ratio. In order to carry out this experiment, we can simulate the data by scaling the same DEM data. We take the data in Fig.4(b) and scale it into different elevation range, shown in Table 3. From Table 3, we can see that the compression ratio becomes low when the elevation range increases.

Table 3  Experiments on same DEM with different elevation range

| Range/m | 0-2000  | 0-1000  | 0-500   | 0-250   |
|---------|---------|---------|---------|---------|
| RMSE/m  | 0.17    | 0.13    | 0.14    | 0.13    |
| Max error/m | 0.8    | 0.6     | 0.7     | 0.6     |
| CR      | 4.62    | 6.31    | 8.18    | 11.16   |

Another experiment, shown in Table 4, is applied on DEM with different resolution in a volcano area. The results demonstrate that the compression ratio is relevant to the resolution of DEM. For the same area, higher resolution DEM allows for higher compression ratio than the lower resolution data because when resolution decreases, the correlation among the data also decreases leading to more efficient compression.

4 Conclusions

This paper introduces an integer wavelet transform to DEM data compression and presents a simple and efficient compression algorithm. When maximal error is limited within 1.0 m, we can derive compression ratio from 4-19 for a variety of DEM data. The compression method discussed in this paper is definitely valuable in DEM applications.

References

[1] Kider D B, Smith D H(2003)Advance in the data compression of digital elevation models[J]. Computers and Geosciences, 29: 985-1002
[2] Cui Baoxia, Duan Yong, Xu Xinhe(2004) Wavelet-based compression of digital elevation models[J]. Information and Control, 33(3): 337-341 (in Chinese)
[3] Chang Zhanqiang, Wu Lixin(2004)Mountain grid DEM data compression base on wavelet transform and mixed entropy coding[J]. Geography and Geo-information Science, 20(1): 24-27 (in Chinese)
[4] Wu Xuewen, Sun Yankui, Tang Long(2004)Lossless compression of DEM based on JPEG2000[J]. Application Research of Computers, 21(1): 240-242 (in Chinese)
[5] Shapiro J M(1993)Embedded image coding using zero-trees of wavelet coefficients[J]. IEEE Transactions on Signal Processing, 41: 3445-3462
[6] Said A, Pearlman W A(1996)A new fast and efficient image codec based on set partitioning in hierarchical trees[J]. IEEE Transactions on Circuits and Systems for Video Technology, 6(3): 243-250
[7] Tian Jun, Wells R O(1998)Embedded image coding using wavelet-difference-reduction[M]/Topiwala P. Wavelet image and video compression. Norwell: Kluwer Academic Publishers
[8] Adams M D, Kossentini F(2000)Reversible integer-to-integer wavelet transforms for image compression: performance evaluation and analysis[J]. IEEE Transaction on Image Processing, 9(6): 1010-1024