2010 International Conference on Electronics and Information Engineering (ICEIE 2010)

An Lidar Data Compression Method Based on Improved LZW and Huffman Algorithm

YiKun Zhang, Xiao Li, DengXin Hua, Hao Chen, HaiYan Jin
Xi'an University of Technology
Xi'an, China E-mail:ykzhang163@163.com

Abstract—Lidar raw echo data has characteristics such as huge data quantity, strong discreteness and unpredictability. According to the construction of Lidar monitoring network of atmospheric environment, the existing network can not provide enough bandwidth to transmit Lidar data in real time. In this paper, we propose a novel hybrid lossless compression algorithm to reduce the transmission amount, namely the probability statistics lossless compression algorithm base on the improved LZW(Lempel-Ziv-Welch), which combines Huffman coding. With experiment on the raw two-value atmospheric data, we verify the effectiveness of our approach that the compression ratio is close to 9.5:1 and the coding efficiency is up to 98%.

Keywords-Lidar data; LZW; Huffman; Hybrid lossless compression method

Ⅰ.INTRODUCTION

Lidar is a new type of active light sensing device, which has developed rapidly in recent years. Because of its high frequency, good performance at low altitude detection and strong anti-interference capability, it has been widely used in many high-tech fields such as environment monitoring, aerospace, communication, navigation and orientation. Especially, it has shown the unique advantages and outstanding prospects in the atmospheric environment monitoring and meteorological measuring [1].

The atmospheric environment monitoring using lidar is a wide range regional activity. It is necessary to establish a monitoring network at region and even nation level used to analyze and share the atmospheric data. However, the amount of the lidar raw atmospheric echo data is very huge. For example, using a lidar with the frequency of 1000HZ to collect 5000 points of vertical height in the continuous sampling period of 5s, the number of sampling data is 2.5× 107. If the storage type is 4 bytes float, the data amount will be 95.37M. The existing network is difficult to transmit so huge data within a short time. With not improving the transmission capacity of the network, data compression technology will be an effective to solve this contradiction.

Generally, data compression can be divided into two categories, namely the lossless and lossy algorithm. For lossy compression, the algorithm based on wavelet transform has been widely used because of its good signal-concentrated nature [2-5]. The lossy algorithm gets a higher compression ratio at the cost of data loss. So it has good effect for the data such as images, audio or video data compressions, which require fewer details. But for the data with more detail, lossless compression method is much applicable. The PPM-based lossless compression algorithm with high efficiency has become a standard of text lossless compression [6, 7]. But it is not suitable for rapid compression and transmission on line because it needs large memory and long running time. The series of algorithms based on LZ(Lempel-Ziv) compression has the characteristics such as low complexity, fast speed and high compression efficiency without prior and statistical knowledge [8]. When only using LZ algorithm to the lidar, the compression ratio is low, usually is between 2: 1 and 5: 1.

According to the above problem, this paper proposes a hybrid lossless algorithm to compress the lidar data, which is referred as the probability statistics lossless compression algorithm based on the improved LZW(Lempel-Ziv-Welch). This algorithm combines LZW compression that belongs to the series of LZ with Huffman coding effectively, and satisfies the requirement of reducing data loss as soon as possible.

Ⅱ.COMPRESSION ALGORITHM DESIGN

The series of LZ compression algorithms reaches close to the entropy rate asymptotically. The longer the length of the string is, the better the compression efficiency is. The codes used in LZ algorithm do not depend on the distribution of signal source, and can be used in the situation without prior knowledge and statistical data. So LZ algorithms are very suitable for the unpredictable lidar atmospheric echo data, and we mainly consider the series of LZ algorithms in our paper.

As one of the LZ algorithms, LZW algorithm is the most representative with low complexity and fast speed. Its output data stream is composed of the order number in the dictionary, not the data itself, which can significantly reduce the length of the output data. However, the Correlation between the data is still large. Considering the characteristics of existing large number of discontinuous repeat feature coding symbols in the output stream, it is possible to compress data further using Huffman coding after initial compression. Then, the increase compression ratio is increased significantly without using extra time. The idea is from the fact that the Huffman coding is based on probability and statistics.

The overall process of the lidar original atmospheric data compression/decompression algorithm proposed in this paper is shown in Fig.1.

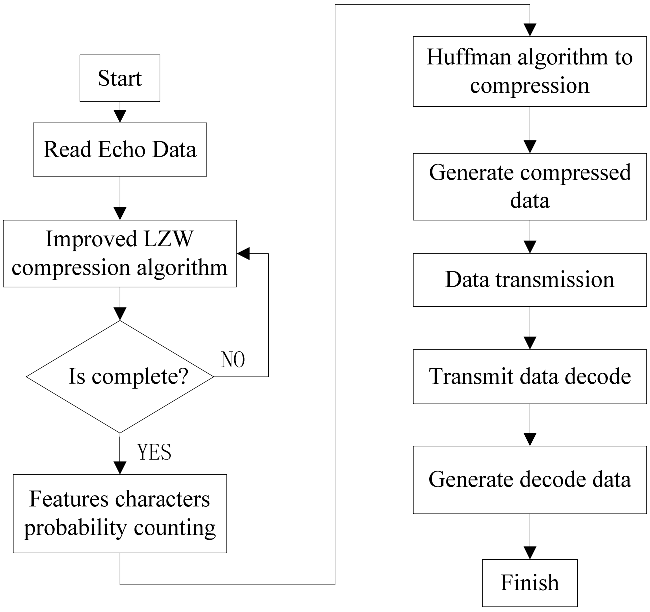


Figure 1. Data compression / decompression flow chart

The compression process includes receiving echo data with the receiver, compressing the data with the improved LZW algorithm, counting the probability of the feature characters, coding data with the Huffman algorithm, finally generating the compressed data and transmitting via the signal channel. The decompression process includes reconstructing the data with the Huffman decoding and the LZW decompression algorithm.

A. Improving the LZW compression algorithm

Experiments show that most time is consumed on searching and matching of the dictionary in LZW compression algorithm. If an effective searching algorithm is built, the time can be reduced to some extent and the compression efficiency is improved. This paper will reduce the searching time in two aspects described in the following sections.

・Reducing the characters when initializing dictionary

Generally, all 256 characters are written into the dictionary in the initialization of the algorithm. Most characters are useless, because only the digital and special separator characters are used in lidar data. According to this obvious feature, if only these necessary feature characters are contained into the dictionary, it will be possible to reduce the dictionary length as well as the searching time.

• Establishing the hash table

Because most time is spent on searching the dictionary, we will improve the search algorithm. There are two algorithms. One is based on the block search that the time complexity is

$$O(\log_2 m + \frac{n}{m})$$

where m represents the number of blocks, and n represents the number of data elements in a block. Another is based on the dynamic tree structure and the time complexity is $O(\log n)$ where n represents node number. These algorithms can reduce the search time to some extent. In this paper, we establish the connection between the searched content and its corresponding storage position based on the hash table. With this connection, we can access the searched content directly. So the time complexity is reduced to $O(1)$, and our algorithm will greatly reduce the searching time when looking in the dictionary.

Fig.2 depicts the comparison result for the different algorithms include the sequential search, block search and our approach. From Fig.2, the efficiency of the sequential search algorithm is very low. The consumed time has reached to 180ms when the data amount is less than 5KB. The block search algorithm improves the search efficiency to some extent, but the time grows too fast as the data amount increases. When the data amount is 12KB, the time is close to 180ms. The growth rate of our approach does not significantly change as the data amount increases. When the data amount is 16KB, the time is only 18.78ms. It is obvious that our approach can greatly increase the searching efficiency and reduce the searching time fundamentally.

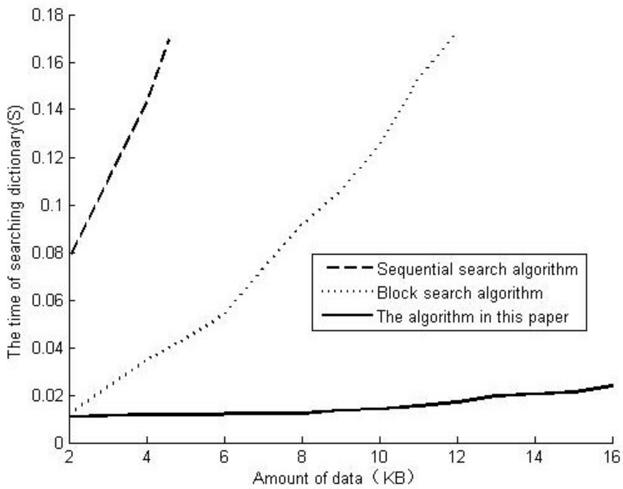


Figure 2. Dictionary searching time of different methods

B. Huffman compression algorithm design

With LZW compression, the number of characters in the output stream is greatly reduced according to the original characters. Now, applying the Huffman algorithm to do the second compression, it will improve the compression ratio with a little additional time. The Huffman compression algorithm is described in the following.

Supposing the source U contains feature character $[a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8]$, firstly the probabilities of feature characters are computed after LZW compression. That is

$$P(X) = \sum_{x_i \in S} P(x_i) \qquad (1)$$

Among which, $P(X)$ represents the probability of a string X in the total character stream, S represents the whole character stream and $P(X)$ represents the probability of each X.

Supposing after the calculation of (1), the probability of each feature character in U is as follows

$$U = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 & a_6 & a_7 & a_8 \\ 0.2 & 0.19 & 0.18 & 0.17 & 0.15 & 0.1 & 0.007 & 0.003 \end{bmatrix}$$

where $a_1 \ldots a_8$ represent each feature character, the corresponding number represents the probability of the feature character.

Because the probabilities of a_8 and a_7 are the least, a_8 and a_7 are designated to "0" and "1", and then the supplementary set is constructed

$$U^{'} = \begin{bmatrix} a_1^{'} & a_2^{'} & a_3^{'} & a_4^{'} & a_5^{'} & a_6^{'} & a_7^{'} \\ 0.2 & 0.19 & 0.18 & 0.17 & 0.15 & 0.1 & 0.01 \end{bmatrix}$$

where $a_k^{'} = a_k$; $k = 1, 2, \ldots, 6$ and $a_7^{'} = a_8 \bigcup a_7$.

Repeating above process until there exists a supplementary set which has only two messages, one is marked with "0" and the other one is marked with "1".

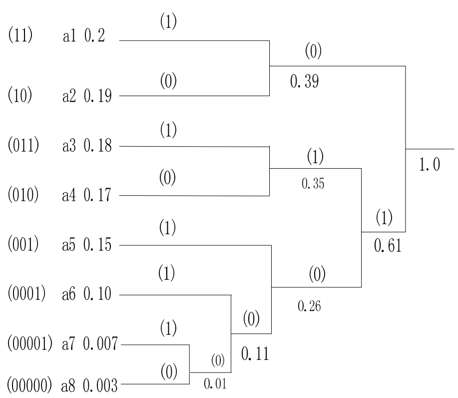


Figure 3. Binary Huffman coding algorithm steps

Fig.3 displays the steps to achieve the binary Huffman coding of source U using the above algorithm, in which the left column is the corresponding code word encoded by the algorithm.

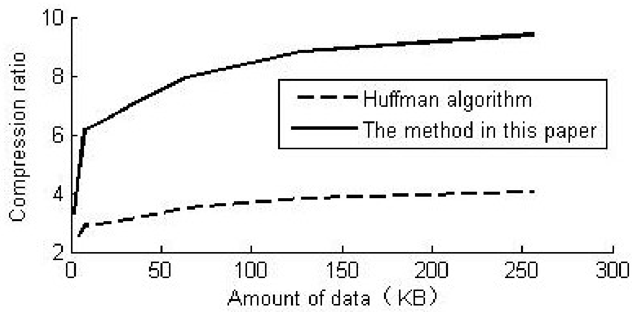
From Fig.3, the average code length R ==2.73(bits / message), the entropy rate of source U is $H(U)$=2.617(bits / message) and the encoding efficiency of Huffman algorithm is computed as $\eta = \frac{H(U)}{R}$=95.8%.

In this paper, the lidar original data firstly is compressed with the improved LZW to reduce the time spent on computing the probability of feature characters. Secondly, the data is re-compressed using the Huffman algorithm to substantially increase the compression ratio at the cost of a small increase of the time. Finally, the coding efficiency is calculated.

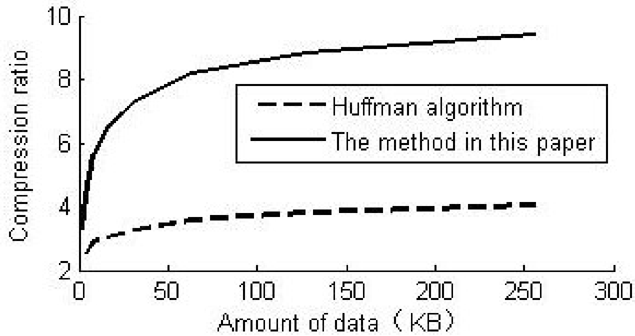
III Experimental Analysis

In our experiments, we compress two set of the Mie scattering lidar atmospheric echo two-value data with our approach and the LZW algorithm respectively. One set consists 2,500 points of the vertical height 0-15Km with the sampling interval of 6m, another consists 5,000 points with the sampling interval of 3m. The experimental conditions include Pentium4 2.4GHz CPU, 1.62GB memory and C# as the programming language in VS 2005. Fig.4 and Fig.5 show the growth curves of the compression ratio and time.

From Fig.4, the actual compression ratio of our approach increases significantly as lidar data increases, and finally is close to 9.5:1. For the LZW algorithm, the actual compression ratio eventually stabilizes at 4:1. To some extent, the sampling interval of the lidar data has little effect on the compression ratio.



（a）Sampling interval of 3 m



（b）Sampling interval of 6 m

Figure 4. Contrast of data compression ratio growth in different data amount between the proposed method and the LZW algorithm

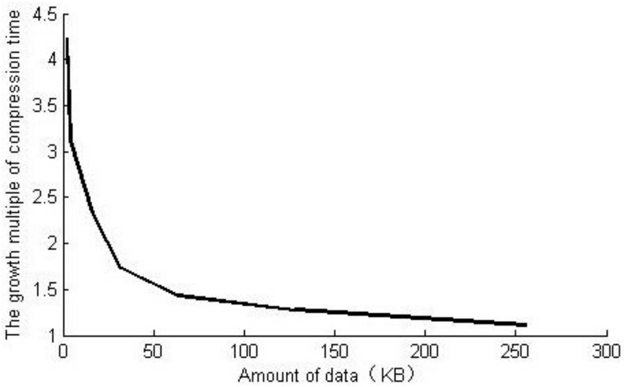


Figure 5. Contrast of data compression time growth between the proposed method and the LZW algorithm

Fig.5 shows that the time growth speed of our approach is reduced significantly with the increase of the data amount, and eventually close to the compression time of LZW.

Table 1 shows the comprehensive performance contrast between the LZW algorithm and our approach according to the data set with sampling interval of 3m.

In Table 1, the compression ratio of our approach is more than twice than the LZW compression algorithm with small increase of compression time. Moreover, for the Mie scattering lidar echo data in a pulse which is about 19.6KB, the compression ratio of the LZW compression algorithm is less than 3:1, and our approach is more than 6:1. For the same amount lidar atmospheric data, if compressed with the Huffman algorithm, the average code length of feature character is R =1 (bits / message), entropy rate $H(U)$ =0.9339(bits / message), the coding efficiency $\eta = \frac{H(U)}{R}$ =93.39%. However, if compressed with our approach, the average code length of feature character is R =3.298 (bits / message), entropyate $H(U)$=3.25 (bits / message), the coding efficiency $\eta = \frac{H(U)}{R}$ =98.55%. Obviously, the average code length of our approach is more close to entropy rate than Huffman algorithm only.

TABLE 1. COMPREHENSIVE PERFORMANCE CONTRAST BETWEEN THE LZW ALGORITHM AND THE PROPOSED METHOD

Compression method / Raw data	LZW Algorithm			The Proposed Method		
	Compression time	Compressed *data* amount	*Compression ratio*	*Compression time*	*Compressed data amount*	*Compression ratio*
2KB	10.92ms	0.86KB	2.326	46.22ms	0.61KB	3.279
4KB	18.24ms	1.62KB	2.469	56.82ms	0.93KB	4.301
16KB	27.42ms	5.53KB	2.893	64.94ms	2.59KB	6.178
19.6KB	38.36ms	6.69KB	2.930	73.69ms	3.08KB	6.364
32KB	54.32ms	10.3KB	3.107	94.14ms	4.61KB	6.941
64KB	93.70ms	18.4KB	3.478	134.10ms	8.07KB	7.931
128KB	194.69ms	33.4KB	3.832	248.26ms	14.51KB	8.828
256KB	433.21ms	63.2KB	4.051	482.61ms	27.19KB	9.412

Ⅳ CONCLUSION

According to the problem existed in the construction of the lidar monitoring network of atmospheric environment such as the existing network cannot provide enough bandwidth to transmit data in real time, this paper puts forward a bybird lossless compression method based on the improved LZW and Huffman algorithm with the requirement of reducing data loss as soon as possible. The experiment results show that, as the increase of data amount, the compression time of the proposed method is close to LZW algorithm, while the compression ratio is significantly higher than the LZW algorithm. At the same time, the average code length is much close to the entropy rate than adopting Huffman algorithm only, and the coding efficiency is much higher than Huffman algorithm. It is indicated that our approach is effective for the lossless compression of the lidar original atmospheric data.

REFERENCES

[1] Qing Yin,JinHai He,Hua Zhang. Application of laser radar in monitoring meteorological and atmospheric environment[J].Journal of Meteorology and Environment,2009.1025(5):48-56.

[2] ZengHui Zhang, WeiDong Hu, WenXian Yu.A Real-Time Radar Video Data Compression Algorithm Based on Lifting Wavelet Transform [J].Journal of Electronics, 2005, 33(10):1910-1913.

[3] Navneet Agrawal，K.Venugopalan. SAR Image Compression using Wavelet Packets [J].2009 International Conference on Signal Processing Systems,IEEE,2009:343–347.

[4] WeiHua Ai,ChaoLing Shen,YunXian Huang. Weather Radar Data Compression with Parallel Adaptive Coding Algorithm[J]. Image and Signal Processing, IEEE, 2008,2:197 – 200.

[5] WeiHua Ai, ChaoLing Shen; YunXian Huang; ZhiGuo Chang. A comparative study of Doppler weather radar data compression[J].Computer Science & Education,IEEE,2009:712 – 715.

[6] J.Cleary，I. Witten. Data compression using adaptive coding and partial string matching, in IEEE Transactions on Communications, 1984,32(4):396-402.

[7] ChangBin Shi.Comression experiment for weather radar reflective data[J].Electronic Test,2010,(1):24-28.

[8] Rizwana Mehboob，Shoab A.Khan，Zaheer Ahmed. High Speed Lossless Data Compression Architecture. Multitopic Conference, 2006. INMIC '06. IEEE,2006:84–88.