

A New Data Compressed Algorithm on Mobile Device under Cloud Service

Zhu Tong

School of Information Engineering
Wuhan University of Technology
Wuhan, China, 13910576010
E-mail: zhutong68@139.com

Gao Zhuo

Computer School
Beijing Information Science & Technology University
Beijing, China, 13601178823
E-mail: gaodigang@bistu.edu.cn

Abstract—In this paper, we discuss how to use cloud services to provide seed data for data compression algorithm on mobile device. Seed data can remove redundant information from GPS data. Then, through the shift encoding algorithm, we provide a more simple and effective data compression method for real-time upload GPS data.

Keywords—cloud services; mobile device; GPS; data compression

I. INTRODUCTION

Currently, cloud computing and mobile computing are used widely on all application. Cloud service is a new kind of IT service that base on cloud computing, the substantive characteristics of cloud service are provide service on demand, pay and play. Cloud service now is the fifth utility that is related with our lives besides Water, Electricity, Natural Gas and Telecommunication, and it will be the new tide of IT[1-3].

Now, Cloud computing platform and mobile applications has been formed to promote each other, mutual promotion, synchronized rapid development trend. On one hand, realization of cloud services will be calculated in the clouds, overcomes the problem of the limited capacity of mobile computing, storage; On the other hand, the characteristics and applications of mobile ubiquitous immediacy, diversity also provides a wide range of needs for the development of cloud computing platforms and development space.

GPS receiver belonging to passive location, location service you want to have to rely on wireless communication system. Most LBS are available through SMS, GPRS, CDMA, and satellite communications to transmit position data. Mobile Terminal will consumes bandwidth and the terminal owns battery when passing up, local temporary storage will consume limited local storage resources, therefore, data compression technology is very important. Although there are many mature methods about data compression, due to the limited computing ability of mobile terminals, data compression algorithms must be simple and effective, with a view to minimizing the consumption of resources on the mobile terminals. On the other hand, in order to reduce the amount of position data transferred, save communication cost, we require a more suitable compression algorithm. For real time data upload, we also need effective solution to the dozens of bytes small amount of data compression issues.

II. PRINCIPLES OF MOBILE GPS DATA COMPRESSION ALGORITHMS UNDER CLOUD SERVICES

Generally we use the Dictionary data compression coding, Huffman coding, predictive coding, wavelet transform method etc[4-13]. to perform GPS data compression, common features of these methods are based on GPS digital sequence's characteristics (which GPS data collected by mobile terminal) to perform data compression. For mobile terminals to access cloud services, cloud services, in particular private cloud services, can based on mobile terminal business features, such as the business area of inspection, data collection, customer service personnel working range, providing seed data, so that you can pass data more then GPS data sequence itself through the seed data in order to gain more data redundancy information to achieve more effective data compression. This is more difficult under the traditional methods. Under the traditional method, mobile terminal itself unable to determine the geographic scope of your business, especially in the mobile terminal users may use the terminal on different regions, but cloud services can be done by provide different business users' geographical scope and relevant information.

Some applications have a very strong time and geographical characteristics, for these applications, through a centralized cloud service to deliver differences seed data to different mobile terminals, like geographical origin, the Center coordinates etc. , mobile terminal then can process and upload compression data (such as GPS coordinates) . That is, removed redundant information in mobile terminal, recoding, upload to a cloud server, cloud service-side decoding based on seed information and data processing. Mobile Terminal's GPS data compression under cloud services architecture as shown in Figure 1.

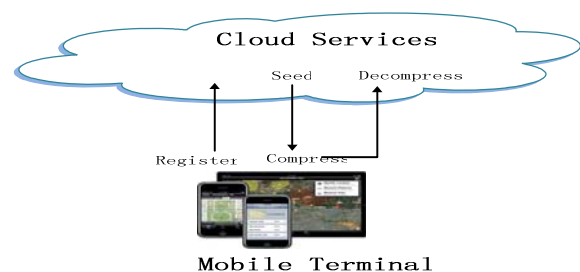


Figure 1 Architecture of Mobile Terminal's GPS data compression under cloud services

III. GPS DATA ANALYSIS AND PRE-PROCESSING

GPS manufacturers follow the United States National Marine Electronics Association NMEA-0183V20 standard form of communication, output data bits 's ASCII codes, which includes latitude, longitude, altitude, speed, date, time, course and a satellite position information, statements include GGA, GLL, GSA, GSV, RMC, VTG, a total of six. For a different purpose, select different statements, for example, in LBS areas such as personal location, vehicle tracking, only concerned with time, longitude, ground speed information, we use RMC records statement. RMC statement has 13 records, including statements identifying tag, world time, position status, longitude, latitude, latitude direction, longitude direction, ground speed, surface routes, dates, magnetic declination, calibration, and an end tag, and all these records typically use about 70 bytes (including 11 comma used to separate records) [5]. Original data obtained from mobile terminals through RS232 serial port. For example:

```
$GPRMC, 130304.998, A, 4717.115, N, 00833.912, E, 00  
0.04, 205.5, 200601, 01.3, W*7C
```

For general inspection, data collection, customer service tracking needs, only requires time, longitude, ground speed and current position status information can be met, additional information can be used as redundant information. This article is only use the needs information for processing.

Data stream received through the RS232 serial port is a text string. The method use in this article is: first of all we do digital processing on a text string, and then via the cloud services based on mobile terminal business characteristics of seed removal of redundant information, complete the compression and decompression. If you do not consider the removal of redundant information, this method belongs to the loss-less compression techniques, the principle is removing redundant bits in the bytes, so as to achieve compression purposes.

According to the study of the original data obtained on the RS232 serial port, can be found between data segments there are a lot of redundancy. For individual applications such as Location, inspection, data collection, customer service, we are concerned only with time, position, status, longitude, ground speed information:

- (1) Universal time (UTC): Information format is: hours, minutes, seconds, milliseconds that indicates that time in the world (converted to Beijing time also need to add 8 hours). Often civilian position data collection is in seconds, so the millisecond level data can be seen as redundant information, after remove MS information, UTC can be express with 6 bytes.
- (2) Position status (A/V): use 1 byte. Because of mobile terminals in some places may not receive the satellite signal, positioning information is invalid.
- (3) Longitude: use 10 bytes
- (4) Latitude: use 9 bytes
- (5) Ground speed: use 4 bytes

- (6)Date: use 6 bytes, displayed in day, month, year format

After passing the above compression pretreatment, retained 6 data records, use a total of 36 bytes. Data structures are shown as follows.

```
struct GPS {  
    CString      strUTCTime;      //UTC      Time,  
    hhmms(HourMinuteSeconds)  
    CString strState; //Position Status, A: Valid data ,  
    V:Invalid data  
    double dlong; // Longitude, A positive value of  
    longitude, negative indicates West longitude  
    double dlat; // Latitude, A positive value of latitude, a  
    negative value indicates South latitude  
    double dSpeed; //Ground speed  
    CString      strUTCDate;      //UTC      Date,  
    ddmmyy(DayMonthYear)  
};
```

IV. COMPRESSION, DECOMPRESSION ALGORITHM

We do further digital processing for the pre-processing GPS data:

- (1) Synthesis to calculate dates and UTC from a basic point of time by integers, for example: C/C++ time function is calculate current dates and UTC from basic point of January 1, 1970, 00: 00: 00 by integers.
- (2) Transfer longitude coordinates, speed to double-precision floating format, complete the digitization of GPS data.

The Data structure as follows.

```
struct DigiCODE_STRUCT {  
    CTime t;  
    unsigned char state;  
    double dlong, dlat;  
    double dSpeed;  
};
```

Cloud server-side based on mobile terminal's business property and area provides seeds for terminal compressing data. By comparing the differential of GPS data and seed values, the terminal removing redundant information, and then begin shift coding to the effective data, complete the compression. Take North China Telecom inspection services in a big city as an example: In space take city minimum coordinates as a seed data, In time adopted January 1, 2005 0 seconds as seed data, after making the difference, you can get the valid time data for 30-bit, valid latitude and longitude data for 21-bit; According to the actual inspection speed, valid speed data limit for 5-bit; Locating State takes a Boolean value so represent by 1 bit.

Then we can take further GPS data after digital compression to 10 bytes. And still reserved 2 bits to use upload mobile terminal key information and other information, GPS compressed data package frame structure as shown in Table 1.

TABLE 1 GPS COMPRESSED DATA PACKAGE FRAME STRUCTURE

bits	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
64	Time(30bits)															
	Longitude(21bits)															
	Latitude high 12 bits												A/V			
16	Latitude low 9 bits									Speed(5bits)				Reserved		

By the shift operation, we can encapsulate effective digital GPS data into a compressed data package frame. Table 2 gives the compression algorithm based on GPS data package per frame (mobile terminals side). Table 3 gives the GPS data package per frame decompression algorithm (cloud services side). Algorithm takes the form of c pseudo-code (code as close as possible to MicroSoft C/C++, can pass C/C++ compile via a small modify). In the code segment, pbuf as a pointer to a data buffer, BaseTime, BaseLong, BaseLat are seed data for time, longitude, latitude. Prefix i64 is for 64 bit certification.

TABLE 2 COMPRESSION ALGORITHM BASED ON GPS DATA PACKAGE PER FRAME

```

i64Long = (__int64)(dlong * 1000000.0);
i64Lat = (__int64)(dlat * 1000000.0);
int iSpeed = (int)(dSpeed);

unsigned __int64 d = 0;
(state=='A')?d |= 0x0000000000000001:d |=
0x0000000000000000;
d |= (i64Time - m_i64BaseTime) << 34;
i64dxLong = i64Long - i64BaseLong;
i64dxLat = i64Lat - i64BaseLat;
d |= i64dxLong << 13;
d |= (i64dxLat >> 9) << 1;
memcpy(pbuf, &d, sizeof(d));
unsigned short c = 0;
c |= i64dxLat << 7;
c |= iSpeed << 2;
memcpy(pbuf + 8, &c, sizeof(c));

```

TABLE 3 DECOMPRESSION ALGORITHM BASED ON GPS DATA PACKAGE PER FRAME

```

(d & 0x00000001)? state='A'; state=='V'
unsigned __int64 d = *((unsigned __int64 *)pbuf);
unsigned short c = *((unsigned short *)pbuf + 8);
CTime t((d >> 34) + m_i64BaseTime);
dlong = (double)(__int64)((d & 0x00000003ffffe000) >> 13) +
i64BaseLong) / 1000000.0;
dlat = (double)(__int64)((((d & 0x00000000000001ffe) << 8) | (c >> 7)) +
i64BaseLat) / 1000000.0;
dSpeed = (double)(int)((c & 0x007c) >> 2);

```

IV. CONCLUSION

Some applications have a very strong time and geographical nature. For these applications, you can use centralized cloud service to delivery difference seed data to different mobile terminals (such as geographical "Origin" or "Center" coordinates), and mobile terminal processed local data (such as GPS coordinates), upload the compressed data. That is mobile terminals removed redundant information, recoded, uploaded to a cloud server-side, cloud server can decoding based on seed information to complete data processing.

This paper analyzes the characteristics of GPS data in the LBS service, through pretreatment and digital for GPS data, then removing redundant information from GPS data with seed, at last by shifting coding to complete data compression. Thus change the raw data (about 70 bytes) to available 36 bytes, then from 36 bytes to 10 bytes through compression. The Main operations can be implemented by and, or and shift operation, can improve coding speed, saving memory resources. Algorithms are based on the frame, has a compression ratio stability, can be applied to dozens of bytes or less small amounts of data to complete effective compression. Through data compression, you can save mobile terminal data transmission bandwidth and communications costs, extend the mobile terminal battery charge time.

Because there are powerful computing resources and performance in the cloud server side, cloud server can provide optimized seed to different mobile terminals applications (for example, inspection, data collection, customer service). For example, can be based on historical data of inspection operations personnel, specify the active route information, and so on, by least square algorithm to generate more optimal seed, further improve the efficiency of compression.

REFERENCES

- [1] Chen Donglin, Ma Mingming, Lü Qiuyun. Cloud Service Collaboration Market Transaction Model [J]. Journal of Wuhan University of Technology (Information & Management Engineering), 2011, 33(3): 456-459. DOI: 10.3963/j.issn.1007-144X.2011.03.032.
- [2] Dong Xiaoxia, Lü Tingjie. Review of the Cloud Computing and Its Future Development [J]. Journal of Beijing University of Posts and Telecommunications (Social Science Edition), 2010, 12(5): 76-81.
- [3] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility [J]. Future generations computer systems, 2009, 25(6): 599-616. DOI: 10.1053/j.gastro.2007.03.107. K. Elissa, "Title of paper if known," unpublished.
- [4] Yang Hongye, Zhang Yue. GPS The Design and Implementation of the Data Compression Algorithm on GPS System [J]. Application of Electronic Technique, 2002, 28(12): 29-32.
- [5] Ren Weizheng, Xu Lianming, Deng Zhongliang. Quasi-losses Compression Algorithm for Commercial GPS Data [J]. Journal of Data Acquisition & Processing, 2010, 25(2): 245-249.
- [6] Zheng Yu, He Dake, Zhang Wenfang. An Efficient Scheme for GPS Data Compression [J]. China Railway Science, 2005, 26(3): 134-138.
- [7] Li Tong, Lü Jun. An Efficient Compressing of GPS Position Data Transmission by Mobile Short Message [J]. Journal of Academy of Armored Force Engineering, 2009, 23(1): 57-59.
- [8] Ren Huaxin. The Design of the Data Compression Algorithm on GPS System [J]. Journal of Changchun Institute of Technology (Natural Science Edition), 2008, 9(3): 74-76.
- [9] Yao Xiaobo, Zhang Jianjun. GPS Data-Compression Based on Wavelet Transform [J]. Journal of Wuhan University of Technology (Information & Management Engineering), 2006, 28(1): 1-3.
- [10] Li Huanbo, Liu Lilong, Chen Lei. Use of Wavelet Function to the GPS Data Compressing Analysis [J]. Geospatial Information, 2009, 7(3): 111-113.
- [11] Shan Xiaoping, Zhao Jing. Wavelet Transform and GPS Data Processing [J]. Jianxi Cehui, 2010, (3): 33-35.
- [12] Piia Saastamoinen, Jari Nurmi, Ilkka Saastamoinen, Mikko Laiho. Minimizing Area Costs in GPS Applications on a Programmable DSP by Code Compression [C]. // 2009 International Symposium on System-on-Chip (SOC 2009). 2009: 91-94.
- [13] Ryan Lever, Annika Hinze, George Buchanan. Compressing GPS Data on Mobile Devices [C]. // On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops pt. 2. 2006: 1944-1947.