# An Image Lossless Compression Patent

Wenyan Wang

College of Electronic Engineering, Guangxi Normal University, Guilin 541004, China Tel: 86-773-582-6559 E-mail: www@mailbox.gxnu.edu.cn

The research is financed by the National Nature Science Fund of China (No. 30973418), the fund of Guangxi Education Department (No. 200808LX019) and the fund of young outstanding teacher of Guangxi Normal University(No.20097). (Sponsoring information)

#### Abstract

The present general lossless compression algorithm is not effective for the compression effect of JPEG files. In this article, the lossless compression method combining the shuffling algorithm with the lossless compression algorithm and a new shuffling algorithm are proposed, and this new algorithm could compresses the JPEG files without losses, and the result indicates that this algorithm can further remove the redundancies in the files, and reduce the volume of the files, and the algorithm has been protected by the patent.

Keywords: Shuffling algorithm, JPEG, Lossless compression

#### 1. Introduction

The picture with the format of JPEG has the character of high image quality, and it is widely used in the computer storage and compute network transfer, and most present pictures of general digital camera are adopt the format of JPEG. Because the picture quality of JPEG picture is higher, so the picture occupies relatively large space, for example, the JPEG picture shot by the digital camera with 8 million pixels can achieve 5M bytes. To save the storage space and reduce the consumption time of picture in the transfer process, the original JPEG picture files are generally compressed to save or transfer.

There are two kinds of compression methods at present. One is the lossless compression which takes the data as the combination of the information and redundancy degree, and its work principle is to remove or reduce the redundancy degree in data without the loss of original data. The other one is the loss compression, which can not only remove or reduce the redundancy degree in data, but ignore some unimportant details, so it has considerable compression ratio, but this compression method will lose some information in original files, so the decompressed files will have some differences with the original files (Yan, 2007).

For the general lossless compression method, the representative algorithms include the Huffman algorithm, LZ and its derived algorithms (LZW, deflate and so on), and the derived software include WinRAR, WinZip, 7-zip and so on, but above software all directly compress the JPEG picture file, so they are not ideal, even the volume of compressed files will increase. In fact, because of the difference of arrangement, original JPEG picture files would have certain redundancy degree which can not be removed when the original JPEG picture files are not treated, so how to more effectively treat the original JPEG picture files is another measure to enhance the compression rate.

A new shuffling algorithm, i.e. the father-common shuffling algorithm, is proposed in the article, and it can effectively treat the original JPEG picture files before these files are compressed, and it could further remove the information redundancy in the files combining with the general lossless compression algorithm. Many experiments show that this method can effectively remove 1%-3% redundancy based on original compression rate, and the algorithm is simple and easy to be realized.

## 2. New shuffling algorithm

The function of the shuffling algorithm is to rearrange the bits of data, and it changes the arrangement of bits into another arrangement, and there are many shuffling functions such as the even shuffling, the k'th sub-shuffling, and so on, and the converse shuffling function could change the arrangement into original arrangement order (Wu, 2006, P.172-175).

The new shuffling algorithm proposed in the article is temporally noted as the father-common shuffling algorithm, i.e. if the former card in front of the card A is the card C, and the former card in front of the card B is also the card C, so the card C is defined as the father card of the card A and the card B. When shuffling, the card

A and the card B are arranged according to the order.

The concrete description of the father-common algorithm combining the JPEG file is showed as follows.

### 2.1 Principle of shuffling algorithm

Treat the JPEG picture files according to the rule of the father-common shuffling.

(1) Read the JPEG picture files by the binary mode, and group these files;

The so-called grouping is to take each 8 bits as one group from the first bit of the binary number, and if there are not 8 bits, 0 can be supplemented in front of this group of binary number to complete 8 bits.

(2) Convert each group of binary code into unsigned decimal number, and save the first group of unsigned decimal number converted from the binary code in the new array H;

Because the unsigned decimal number converted from the first grouping number is in 0-255, and to save spaces, this unsigned decimal number can be set as the space occupying one byte.

- (3) Sequentially establish 256 arrays Ei (i=0-255), and these arrays are used to store the sub-data after the father-data corresponding with i. For any two neighboring groupings, the former grouping data is set as the father-data, and the latter grouping data is set as the sub-data;
- (4) Traverse all converted unsigned decimal numbers according to the order of ascending counts, and store the sub-data occurring after each father-data in the i'th array E[i] established in the step (3), and i=father-data, and if the father-data same with i doesn't exist, the array E[i] is empty;
- (5) Judge whether the traversing ends, and if it ends, count the length of each array that the traversing is completed, and store the result in the new array G, and if it doesn't end, continue the step (4);

The element which records the length of each array in the array G generally occupies the space of four bytes.

- (6) Connect the contents in 256 Ei arrays end to end and save them in the new array I;
- (7) Connect and combine the contents in the array H, the array G and the array I end to end, and store them in the new array J which is the converted JPEG picture file.

It is seen that the volume of converted files is bigger than the original files for 1001 bytes.

## 2.2 Reverse-shuffling algorithm

Implement the reversely shuffling to the converted JPEG picture files.

(1) Read reversibly transformed JPEG picture files by the binary mode in the new array J, and group the data;

The grouping begins from the first bit, and each 8 bits generally is one group.

(2) Convert each group of data into unsigned decimal numbers, and read the first unsigned decimal number and store it in the new array H, and read the length element of the array stored in sub-data, and store this length element in the new array G sequentially;

Sequentially read the spaced occupied in the data according to the above rules, i.e. read the unsigned decimal number occupying 1 byte in the array J into the array H, and read the array length with four bytes after the unsigned decimal number into the array G.

- (3) Sequentially establish 256 arrays Ei (i=0-255), and they are used to store the sub-data after the father-data corresponding with i;
- (4) Sequentially store the residual data in the array J according to the length of the sub-data array in the array G in the array E[i] established in the step (3);
- (5) Establish the array A, and store the data in the array H in the first position of the array A;
- (6) Sequentially read the data in the array A as the father-data, and look for the first number without read sign in the array E[i] (i=father data) in 256 arrays as the sub-data, and store this sub-data in the next empty position after the father-data in the array A, and sign the position of this sub-data in the corresponding array E[i] as read;
- (7) Scan 256 arrays Ei, and check whether the sign positions of all data is read, and if they are not read, so continue the step (6), and if they are read, so stop scanning, and save the array A as file, and the file is the original JPEG picture file.

#### 2.3 Flow chart

The compression/decompression flows are seen in Figure 1 and Figure 2.

## 3. JPEG files lossless compression/decompression steps based on the father-common shuffling algorithm

- 3.1 Compression steps
- (1) Reversibly convert the JPEG picture file according to the father-common shuffling rule, and obtain the converted JPEG picture file;
- (2) Compress the converted JPEG picture files by the LZ77 coding or the improved algorithm, and the compression is over.
- 3.2 Decompression steps
- (1) Decompress the compressed JPEG picture files by the LZ77 coding or the improved algorithm, and obtain the reversibly converted JPEG picture files;
- (2) Reversely treat the reversibly converted JPEG picture files by the reverse shuffling algorithm, and obtain original JPEG picture files.

## 4. Experiment

Experiment environment: Duo 2 T9300 computer, Windows XP.

Experiment data: Canon EOS 400D Digital, general JPEG pictures and network pictures shot by Cannon PowerShot D630 camera.

Experiment tool: general gzip lossless compression algorithm.

Part experiment result is seen in Table 1.

# 5. Result analysis

Volume analysis: Above results are only part data, and large numbers of experiment data show that this method can compress the volume of JPEG files without loss, and when the file is the JPEG picture over 1M bits, the compression effect is better than the current compression software.

Real-time analysis: In above experiment environment, to compress the single JPEG file with 20M bits, Winrar needs about 14s, and Winzip needs about 3s, and the father-common shuffling algorithm +gzip lossless compression method needs about 5.5s.

To the data in the Table, there are 40 JPEG picture files, and Winrar needs 72s, Winzip needs 12s, and the father-common shuffling algorithm +gzip lossless compression method needs about 40s.

#### 6. Conclusions

At present, the picture of JPEG format is popular and widely used in various occasions, and its volume gradually becomes into the restriction of transfer and saving with the development of the digital equipments. Under the condition to possibly reduce the volume without the loss of picture quality, a new shuffling algorithm combining with the general lossless compression algorithm is proposed in the article from another angle. The experiment result indicates that the real-time character of this method is good, and the compression rate can be enhanced by 1%-3% comparing with WinRAR, WinZip and 7zip, and the concretely theoretical support is still in research, but the algorithm has been protection by the patent (No. 200810073769.0).

#### References

Wu, Guoqing & Chen, Hong. (2006). A Lossless Compression Scheme for Scientific Data from Simulation. *Computer Engineering and Applications*. No.5. P.172-175.

Yan, Jingwen. (2007). Digital Image Processing (Matlab Edition). Beijing: National Defense Industry Press.

Table 1. Comparison of JPEG pictures (shot by Cannon EOS 400D Digital Camera) respectively compressed by Winrar, Winzip and father-common+gzip compression algorithm

Original file (bits)	Winrar (compressed)	Winrar (compression rate)	Winzip (compressed)	WinZip (compression rate)	Father-common +gzip	Father-common +gzip (compression rate)
3681869	3681869	1	3677638	0.998850855	3638128	0.988119892
3770812	3770812	1	3766598	0.998882469	3726476	0.98824232
3945319	3945319	1	3941177	0.998950148	3899604	0.988412851
3417356	3417356	1	3412817	0.99867178	3377434	0.988317869
3835407	3835407	1	3831230	0.998910937	3789818	0.988113647
4475246	4475246	1	4471177	0.999090776	4419403	0.987521803
4792901	4792901	1	4788787	0.999141647	4729471	0.986765844
5384009	5384009	1	5378453	0.998968055	5301398	0.984656229
5449415	5449415	1	5443992	0.999004847	5368324	0.98511932
4626088	4626088	1	4622023	0.999121288	4564513	0.986689618
3565861	3565861	1	3561495	0.998775611	3520582	0.987302085
4293851	4293851	1	4290416	0.999200019	4242200	0.987970938
4550468	4550468	1	4546345	0.999093939	4492512	0.987263728
3460901	3460901	1	3456472	0.998720275	3420616	0.988359968
3536069	3536069	1	3531752	0.998779153	3495277	0.988464026
3558762	3558762	1	3554446	0.998787219	3518580	0.988708995
3717969	3717969	1	3713717	0.998856365	3677408	0.989090549
5282652	5282652	1	5278222	0.999161406	5206423	0.985569937
4376971	4376971	1	4372845	0.999057339	4324703	0.988058408
4570808	4570808	1	4566708	0.999103003	4513925	0.987555154
4055039	4055039	1	4050874	0.998972883	4007676	0.988319964
3916298	3916298	1	3912733	0.999089702	3874502	0.989327676
3685686	3685686	1	3681444	0.998849061	3642839	0.988374756
3919988	3919988	1	3915819	0.998936476	3874766	0.98846374
4181715	4181715	1	4177595	0.999014758	4133712	0.988520739
4304162	4304162	1	4300016	0.999036746	4254864	0.988546435
4427687	4427687	1	4423583	0.999073105	4371645	0.987342827
4418464	4418464	1	4414379	0.999075471	4362789	0.987399467
4761633	4761633	1	4757281	0.999086028	4698556	0.986753074
4444537	4444537	1	4440430	0.999075944	4388683	0.987433112
4576409	4576409	1	4572229	0.99908662	4518208	0.987282387
4223968	4223968	1	4219772	0.999006621	4174354	0.988254172
4111795	4111795	1	4107583	0.99897563	4063590	0.98827641
4865350	4865350	1	4860788	0.999062349	4802097	0.986999291
5053803	5053803	1	5048597	0.998969885	4983078	0.986005588
4961269	4961269	1	4956484	0.999035529	4890849	0.985806051
4484859	4484859	1	4480780	0.999090495	4429392	0.987632387
4178102	4178102	1	4173947	0.999005529	4125242	0.987348322
4757874	4757874	1	4754066	0.999199643	4695201	0.98682752
5041017	5041017	1	5035839	0.998972826	4972104	0.986329544

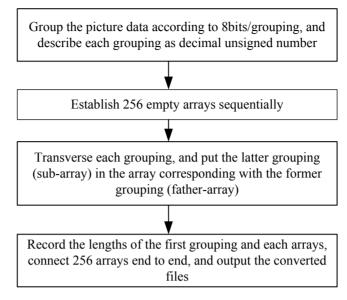


Figure 1. Flow of Shuffling Processing of JPEG Picture File

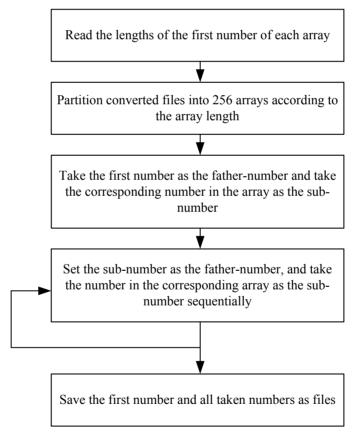


Figure 2. Flow of Reduction Treatment of Decompressed JPEG Picture File