Improved JPEG based GPS picture compression

Yair Wiseman[,1]

[1] Computer Science Department, Bar-Ilan University, Ramat-Gan 52900, Israel
wiseman@cs.biu.ac.il

Abstract. Pictures used by GPS devices are usually overly large to be stored as Bit Map, so they are by and large compressed. The common compression method is JPEG; however JPEG assumes that the average color of beginning of each blocks line is usually similar to the average color of end of its preceding blocks line. Many times this assumption is incorrect for GPS picture. This paper suggests a way to change this method to an improved method that is more suitable for JPEG based GPS images.

Keywords: Image Compression, JPEG, GPS.

1 Introduction

Many GPS application use maps stored in small squares format [1]. These small squares are stored in the well known JPEG format. JPEG compression algorithm performs these operations:

At first, it transforms the image color into a suitable color space. The most common procedure is the converting into YUV components. These components are interleaved together within the compressed data. The ratio between these components is usually not one to one. The Y components will usually have a four times weight. The human eye is less sensitive to the frequency of chrominance information than to the frequency of luminance information which is represented by the Y component in the YUV format. Hence, the Y component gets a higher weight [2]. i.e. the order is 4 Y blocks, one U block, one V block and then over and over again Y blocks, one U block, one V block; whereas each block contain 8X8 pixels.

The order of the blocks is line by line and each line is read from left to right. After the group into blocks, JPEG transforms each block through a Forward Discrete Cosine Transform (FDCT) [3]. The DCT gives a frequency map, with 8X8 or 64 elements. The transformation keeps the low frequency information which a human eye is sensitive to. Subsequently a quantization is done. In each block, each of the 64 coefficients is divided by a separate "quantization coefficient". The quantization coefficients are set according to the desired image quality. In point of fact, the image creator decides about a quality level (which is a number amongst 1 to 100) and according to this number the quantization coefficients are set. There is no standard method how to translate this quality level number into the quantization coefficients,

so the quantization coefficients themselves are stored in the compressed image. The results of the division are rounded to integers. This step loses some information because of the rounding. Furthermore, it can be noted that even if the quantization coefficient is 1, some information will be lost, because more often than not the DCT coefficients are real numbers.

Finally, JPEG uses a Huffman compression method with the aim of reducing the size of the data. JPEG encodes the reduced coefficients using Huffman codes (Arithmetic coding is also supported, but it is not commonly used). There are also other compression algorithms that fit JPEG, but are not officially supported [4].

2 Improved JPEG based GPS picture compression

As was mentioned above the blocks of JPEG are stored line by line. So, the first block of line no. N+1 is stored after the last block of line no. N. Accordingly, the order of the block is as depicted in Figure 1:
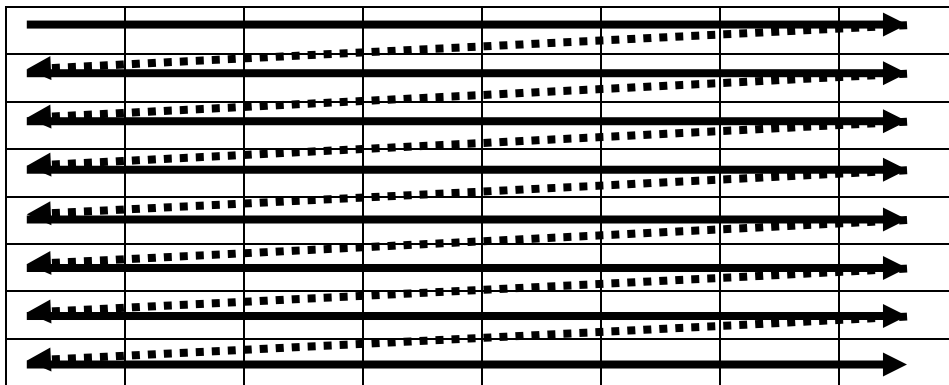

Fig. 1. The order of the lines in JPEG image

The first value of each block is "DC". DC is the average intense of the block multiplied by 8. It should be noted that DC is not stored as a plain value; rather it is stored as the difference between the current DC value and the previous DC value [5]. Usually, the DC values of adjacent blocks is similar [6,7], so it is commonsensical to store the difference which is a smaller number instead of the original number [8,9,10]; however the difference between the last DC in a line and the first DC in the next line is not always small because there are not DC values of adjacent blocks.

Figure 2 is a GPS map of a region in Israel. Each line of image's blocks is of 16 pixels height. Each block is 8X8 pixels, but since for each four Y block there are only one U block and one V block each four pixels of U, just one pixel is actually stored and its value is the average of the four pixels. The same is for V block. So for each 16X16 block of pixels there are four Y blocks, one U block and one V block.

In Figure 3 we can see the first and the second 16 pixels lines of the image. The average DC values of the last blocks of the first line and the average DC values of first block of the second line are shown in Table 1.
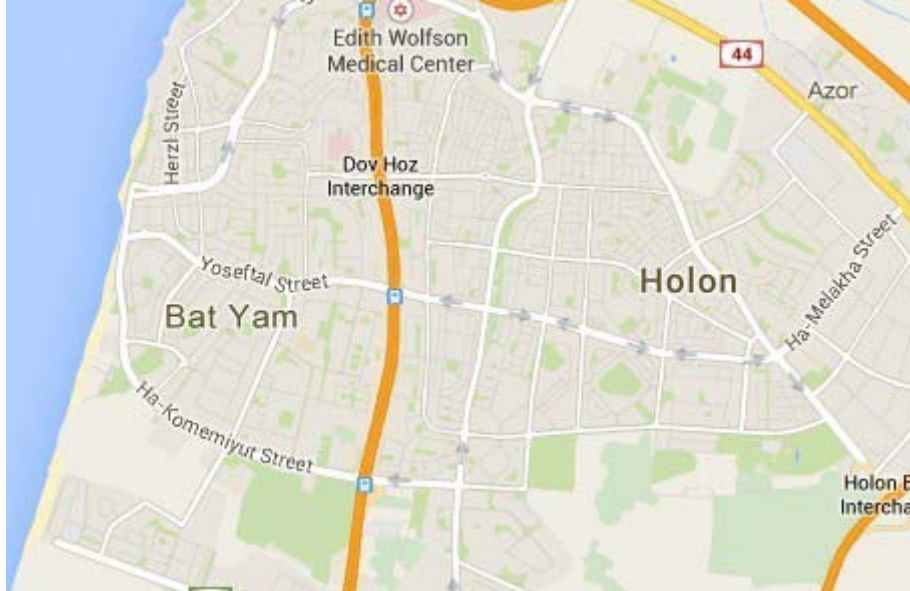
Fig. 2. Example of GPS map.

As can be easily seen the average colors of the beginning of the lines is usually very similar and sometime is even identical. The likelihood for the average colors of the beginning and the end of the lines to be similar is to a great extent lower [11,12]. In particular, the differences between the average colors of the beginning of the lines in Figure 3 for Y, U and V are all zero i.e. they are totally identical. The DC values of Y, U, V in the end of line 2 are 237, 125, 127, accordingly the differences between the average colors of the end of the lines in Figure 3 for Y is 1, for V is 3 and for V is also 3. In view of the fact that the DC values are stored as the differences between the current value and the previous value, we should focus on the differences. When there is no difference, according to [13], it will be indicated by JPEG by the sequence 00 which is a two bits string; however, 32 will be indicated by the string 1110100000 and 34 will be indicated by 1110100010 which are both 10 bits strings.

For that reason, we suggest to change the order of the blocks in JPEG images to the order in Figure 4.

Fig. 3 Lines of 16 pixels.

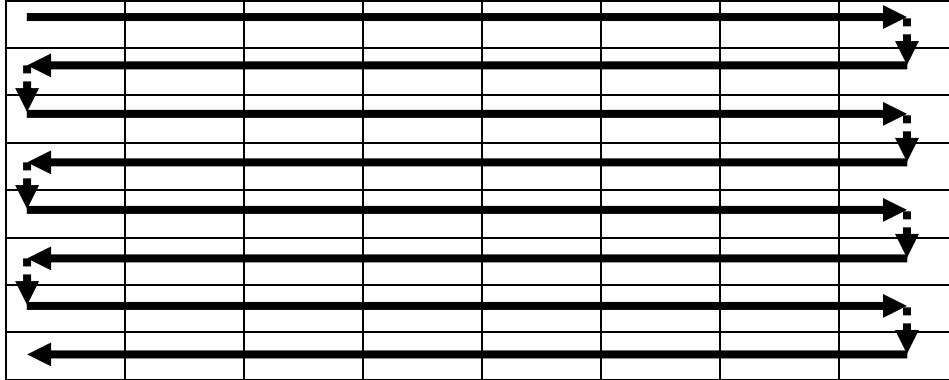|	First line DC	Second line DC	Difference
Y	236	204	32
U	122	156	34
V	130	108	22

Table 1. DC values.

Fig. 4. New order line for JPEG images.

With the new order the difference between the DCs of adjacent lines will be significantly smaller and the image will be compressed better.

3 Conclusions

GPS memory complicatedness [14] makes compression essential [15]. A technique of compression for JPEG based GPS picture with a higher compression ratio has been introduced. The results are encouraging. The difference between the values of each blocks line can be dramatically decreased and as a result a much small compressed image can be obtained.

References

1. L. Cao, J. Krumm, "From GPS traces to a routable road map", Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 3-12, Seattle, Washington, 2009.
2. S. H. Lee and I. C. Nam, "Intra prediction method based on the linear relationship between the channels for YUV 4:2:0 intra coding", 16th IEEE International Conference on Image Processing (ICIP- 2009), pp. 1037-1040, Cairo, Egypt, 2009.
3. Y. Wiseman, "The still image lossy compression standard – JPEG", Encyclopedia of Information and Science Technology, Third Edition, Vol. 1, Chapter 28, 2014.
4. Y. Wiseman, Burrows-Wheeler Based JPEG, Data Science Journal, Vol. 6, pp. 19-27, 2007.
5. S. T. Klein and Y. Wiseman, "Parallel Huffman Decoding with Applications to JPEG Files", The Computer Journal, Oxford University Press, Swindon, UK, Vol. 46(5), pp. 487-497, 2003.
6. C. Y. Lin and Chang S. F., "A robust image authentication method distinguishing JPEG compression from malicious manipulation" , IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, no. 2, pp. 153-168, 2001.

7. G. K. Wallace, "The JPEG still picture compression standard." IEEE Transactions on Consumer Electronics, Vol. 38, no. 1, pp. 18-34, 1992.
8. Y. Wiseman, "Take a Picture of Your Tire!", Proc. IEEE Conference on Vehicular Electronics and Safety (IEEE ICVES-2010) Qingdao, ShanDong, China, pp. 151-156, 2010.
9. Y. Wiseman, "The Effectiveness of JPEG Images Produced By a Standard Digital Camera to Detect Damaged Tyres", World Review of Intermodal Transportation Research, Vol. 4(1), pp. 23-36, 2013.
10. Y. Wiseman, "Camera That Takes Pictures of Aircraft and Ground Vehicle Tires Can Save Lives", Journal of Electronic Imaging, Vol. 22(4), 041104, 2013.
11. E. Fredj and Y. Wiseman, "An O(n) Algorithm for Edge Detection in Photos Compressed by JPEG Format", Proc. International Conference on Signal and Image Processing SIP-2001, Honolulu, Hawaii, pp. 304-308, 2001.
12. Y. Wiseman and E. Fredj, "Contour Extraction of Compressed JPEG Images", ACM - Journal of Graphic Tools, Vol. 6(3), pp. 37-43, 2001.
13. B. W. Pennebaker and J. L. Mitchell, "JPEG: Still image data compression standard", Kluwer Academic Publisher, Norwell, MA, 2004.
14. D. Livshits and Y. Wiseman, "The Next Generation GPS Memory Management", International Journal of Vehicle Information and Communication Systems, Vol. 3(1), pp. 58-70, 2013.
15. R. Steinmetz, "Data compression in multimedia computing - principles and techniques" Multimedia Systems, Vol. 1, no. 4, pp. 166-172, 1994.