

Alleviation of JPEG Inaccuracy Appearance

Yair Wiseman

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

Abstract

JPEG images are almost always self-synchronized after an occurrence of an error in the JPEG file. In spite of the self-synchronization feature, after the synchronization point in the image, the rest of the image is not shown as it has been in the genuine image. Commonly, the blocks are shifted to right or left and usually the tinge is damaged as well. This paper suggests a way how to remedy the block shift and the tinge damage.

Keywords: Image Compression, JPEG, GPS

1. Introduction

In the header of each JPEG image the size of the image is encoded as X, Y whereas X and Y are integers defining the width and the height of the image respectively [1]. Unlike faxes that have a special code for end of line [2], JPEG pictures will start the next line right after the previous line comes to an end. The X value stored in the JPEG header file is the only indication for beginning of a new line. There is no codeword stands for end-of-line.

Any type of JPEG files contains several markers [3-4]. Baseline JPEG files (which is the most common JPEG format in use) have a marker indicating the beginning of the image data. This marker is the sequence of two characters 0xff and right after that 0xc0. Other JPEG formats have other markers for that.

After this marker the following information appears in accordance with this order:

Three bytes which are not related to this paper.

Two bytes containing the image height - Y (in pixels).

Two bytes containing the image width - X (in pixels).

More information unrelated to this paper.

JPEG images are split into block of 8X8 pixels [5-6]. An error in X or Y can produce a damaged image as described in Figure 1a and Figure 1b. This Figure contains an image with an error in the X value. The X value was increased by eight, so each line is longer by one block of 8X8. This increase caused a surplus of one block for each line, but it is accumulated for several blocks. *i.e.* the second line has been pushed left by one block, the third line has been pushed left by two blocks and as a general rule line N was push by N-1 blocks.

However, most of the damages are in the rest of the file. They automatically synchronized [7-8]; nevertheless, they can cause various damages:

An erroneous block (or even several blocks) can be mistakenly added and this block will push the line containing this block to the left and as a result all the rest subsequence lines will be pushed to the left as well.

A block (or even several blocks) can be incorrectly deleted and this will pull the line containing this block to the right and as a result all the rest subsequence lines will be pulled to the right as well.

The average tinge of each block is stored in the beginning of each block. This value is called DC [9]. The DC is not stored as an absolute value, but rather as a relative value [10-11]. *i.e.* the DC stored in each block is the difference between the current DC value

and the DC value of the previous block. A DC value can be incorrect as a result of an error. In such a case, the tinge of this block can be incorrect, but since the values of all the DCs are stored as differences, one wrong DC value will trigger incorrect DC values for all the rest of the blocks in the image.



Figure 1a. Original Undamaged JPEG File



Figure 1b. JPEG File with an Incorrect X Value

Figure 2a and Figure 2b demonstrate these problems. Figure 2a contains the original correct image, whereas Figure 2b contains an image with an error. There was an additional block near the jet engines in this image. This additional block causes two problems:

The additional incorrect block pushed the block of its line to the left and as a result all the lines in the rest of the image were pushed left as well as can be seen in the bottom of the image.

The additional incorrect block had a substantial difference between its DC value and the DC value of the previous block. Therefore, a large number was reduced from the DC value and this reduction made the bottom part of the picture very bright.

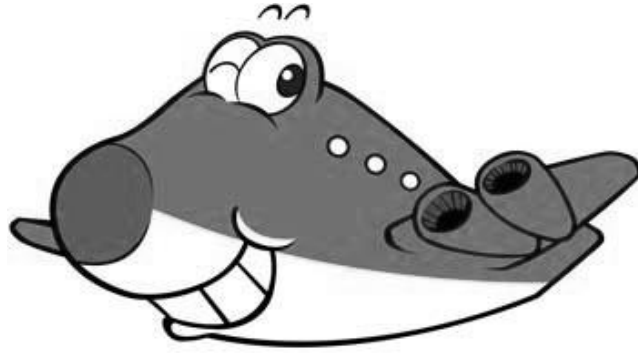


Figure 2a. Original Undamaged JPEG File



Figure 2b. Incorrect Placing of Blocks and Incorrect Tinge because of an Error

Another error that can happen is repositioning of the image portion below the error point from right to left or from left to the right *e.g.* the sword in Figure 3a was repositioned in the left side of the image in Figure 3b. In this image, several additional erroneous blocks have been added because of the error; therefore the blocks have been pushed to the right, but there has been no place for them in the end of the line. As a result they have been repositioned to the beginning of the next line *i.e.* the left side of the image.



Figure 3a. Original Undamaged JPEG File



Figure 3b. Incorrect Repositioning of Blocks and Incorrect Tinge because of an Error

2. How to Alleviate JPEG Inaccuracy Appearance

Some compression formats like TIFF have an "end of line" codeword [12]. Such a codeword can help JPEG to know where the real end of line is and the lines will not be moved as described in Figure 1a, Figure 1b, Figure 2a and Figure 2b. Indeed, an end-of-line codeword have two major disadvantages:

It takes some space to store end-of-line codewords for all the lines and makes the compression less efficient; however, some of the efficiency loss can be recompensed by eliminate the need for use of the X and Y components in the header.

JPEG usually employs Huffman as its entropy encoder although some other compression techniques are available [13]. A new codeword will make other Huffman codewords of JPEG longer. Therefore, this additional end-of-line codeword will also make the compression less efficient.

However, the less efficiency of the compression will be compensated by better images in case of errors, particularly when the communication lines are noisy and there is a realistic possibility for an error in the transfer. Also, critical safety equipment using JPEG files like [14,15,16] may necessitate more accurate and proper data. Furthermore, some of the data in JPEG files is redundant anyway and is sometimes replaced by steganography systems in order to hide data [17,18,19], so if the data does not attempt to be the best compressed, some more bits for an end-of-line codeword will not make a significant harm.

As was mention above, the DCs of JPEG are stored as differences. This is again better for compression efficiency. The differences are usually quite small and it is more efficient to store small numbers rather than longer numbers; however, storing the DC values themselves will help JPEG to show much more accurate images. If the DC values themselves were stored in the JPEG file, the bright bottom part of the image in Figure 2b would have been shown in the original tinge as it is shown in Figure 2a with no tinge change.

Yet again, storing the DC values themselves is trading off with compression efficiency; however, the less efficiency of the compression will be compensated by better images in case of errors [20].

3. Progressive Mode

JPEG has a default format called "baseline JPEG" [21]. In baseline JPEG format the blocks are encoded in a sequential order. In addition to the baseline JPEG format, JPEG also support some progressive modes [22].

The progressive modes are intended to support real-time transmission of images [23]. In these modes there are several scans of the image and with each scan, the decoder will be able to produce a higher-quality rendition of the image. Therefore, a low-quality preview can be sent very quickly, then refined if the user is still interested in the image and as time allows.

The total space needed to store an image in one of the progressive mode is roughly the same as for a baseline JPEG image of the same quality [24] as the objective is not better compression but rather facilitating real-time transmission of images.

The notable disadvantage of the progressive modes is the need of a buffer for the decoder. Since the information does arrive sequentially, it must be stored for the upcoming refinements.

The most common JPEG progressive mode is Spectral Selection. In Spectral Selection mode, in the beginning, the DC values are sent. Subsequently, the AC values are sent according to their frequencies. The low frequency AC values are sent first followed by the high frequency AC values. When just DC values are arrived, each 8x8 block will be filled with equal pixels and the image will be "pixilated". When more and more AC values arrive, the image will become clearer.



Figure 3a. Original Image in Spectral Selection Mode



Figure 3b. An Image in Spectral Selection Mode with an Error in the DC Scan

When there is an error in Spectral Selection mode, one of the scans can be corrupted. If the DC scan is corrupted, the DC values from the error point can be incorrect, whereas the AC values remain unchanged. In such cases the DC values might be shifted to the left or the right and their color can be inaccurate, whereas the AC values will be correct and therefore the outline of the image will be as it should be.

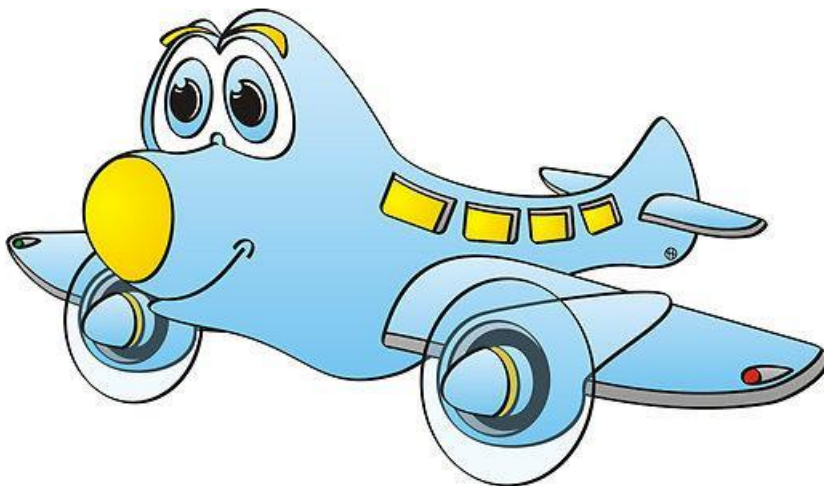


Figure 4a. Original Image in Spectral Selection Mode



Figure 4b. An Image in Spectral Selection Mode with an Error in the DC Scan

Figure 3a and Figure 3b depict such a case. Figure 3a is the original image in Spectral Selection mode. In Figure 3b the same image has an error in the DC scan. This error caused a shift to the left of all the block colors in the lower part of the image. The AC values however, have not been shifted, so the outline of the image in the lower part of the image is correct.

In Figure 3b there is no noticeable error in the global colors of the lower part of the image; however in some cases such an error can occur. Figure 4a and Figure 4b show a case where an error in the DC scan caused an error in the global colors of the lower part of the image as well as the error of shifting of the block color to the right.

5. Conclusion

Errors in JPEG images are almost always synchronized; however, the images after the synchronization are not so clear and sometimes for essential apparatus using JPEG [25-26], the data is required to be much more correct.

This paper presents techniques how to resolve the problems of incorrect appearance of JPEG images after the synchronization point so as to make the images clearer and more comprehensible.

References

- [1] W. B. Pennebaker and J. L. Mitchell, "JPEG: Still image data compression standard", Springer Science & Business Media, (1993).
- [2] R. Buckley, D. Venable and L. McIntyre, "New developments in color facsimile and internet fax", In proceedings of Color and Imaging Conference, Society for Imaging Science and Technology, vol. 1997, no. 1, (1997), pp. 296-300.
- [3] Y. Wiseman, "The still image Lossy compression standard – JPEG", Encyclopedia of Information and Science Technology, Third Edition, vol. 1, no. 28, (2014).
- [4] G. K. Wallace, "The JPEG still picture compression standard", Communications of the ACM, vol. 34, no. 4, (1991), pp. 30-44.
- [5] Y. Wiseman and E. Fredj, "Contour Extraction of Compressed JPEG Images", Journal of Graphic Tools, vol. 6, no. 3, (2001), pp. 37-43.
- [6] E. Fredj and Y. Wiseman, "An O(n) Algorithm for Edge Detection in Photos Compressed by JPEG Format", Proceedings of International Conference on Signal and Image Processing SIP-2001, Honolulu, Hawaii, (2001), pp. 304-308.
- [7] S. T. Klein and Y. Wiseman, "Parallel Huffman Decoding with Applications to JPEG Files", The Computer Journal, Oxford University Press, Swindon, UK, vol. 46, no. 5, (2003), pp. 487-497.

- [8] S. T. Klein and Y. Wiseman, "Parallel Huffman Decoding", Proceedings of Data Compression Conference, DCC-2000, Snowbird, Utah, USA, (2000), pp. 383-392.
- [9] A. C. Hung and T. H. Y. Meng, "Optimal Quantizer step sizes for transform coders", In IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-91, pp. 2621-2624.
- [10] Y. Wiseman, "Enhancement of JPEG Compression for GPS Images", International Journal of Multimedia and Ubiquitous Engineering, vol. 10, no. 7, (2015), pp. 255-264.
- [11] Y. Wiseman, "Improved JPEG based GPS picture compression", Advanced Science and Technology Letters, vol. 85, (2015), pp. 59-63.
- [12] T. Davenport and M. Vellon, "Tag Image File Format (TIFF)-Rev. 4.0", Aldus Corp. & Microsoft Corp, (1987).
- [13] Y. Wiseman, "Burrows-Wheeler Based JPEG", Data Science Journal, vol. 6, (2007), pp. 19-27.
- [14] Y. Wiseman, "Take a Picture of Your Tire!", Proceedings of IEEE Conference on Vehicular Electronics and Safety (IEEE ICVES-2010), Qingdao, Shan Dong, China, (2010), pp. 151-156.
- [15] 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, no. 1, (2013), pp. 23-36.
- [16] Y. Wiseman, "Camera That Takes Pictures of Aircraft and Ground Vehicle Tires Can Save Lives", Journal of Electronic Imaging, vol. 22, no. 4, (2013), pp. 41-104.
- [17] Y. Wiseman, "Protecting Seaport Communication System by Steganography Based Procedures", International Journal of Security and Its Applications, Sandy Bay, Tasmania, Australia, vol. 8, no. 4, (2014), pp. 25-36.
- [18] Y. Wiseman, "Steganography Based Seaport Security Communication System", Advanced Science and Technology Letters, vol. 46, (2014), pp. 302-306.
- [19] J. Fridrich, "Feature-Based Steganalysis for JPEG Images and its Implications for Future Design of Steganographic Schemes", In Information Hiding, Springer-Verlag, Berlin Heidelberg, (2005), pp. 67-81.
- [20] Y. Wiseman, "Diminution of JPEG Error Effects", Advanced Science and Technology Letters, (2015).
- [21] P. S. Sindhu, J. M. Frailong, D. J. Curry, A. Nafarieh and D. Kletter, "Adaptive Quantization Compatible with the JPEG Baseline Sequential Mode", U.S. Patent 6,175,650, issued January 16, (2001).
- [22] C. B. Ahn, I. Y. Kim and S. W. Han, "Medical image compression using JPEG progressive coding", In IEEE Nuclear Science Symposium and Medical Imaging Conference, (1993), pp. 1336-1339.
- [23] T. S. Chen and C. Y. Lin, "A new improvement of JPEG progressive image transmission using weight table of quantized DCT coefficient bits", In Advances in Multimedia Information Processing, Springer, (2002), pp. 720-728.
- [24] K. Taekon, H. M. Kim, P. S. Tsai and T. Acharya, "Memory efficient progressive rate-distortion algorithm for JPEG 2000", IEEE Transactions on Circuits and Systems for Video Technology, vol. 15, no. 1, (2005), pp. 181-187.
- [25] Y. Wiseman, "Device for Detection of Fuselage Defective Parts", Information Journal, Tokyo, Japan, vol. 17, no. 9, (2014), pp. 4189-4194.
- [26] Y. Wiseman, "Fuselage Damage Locator System", Advanced Science and Technology Letters, vol. 37, (2013), pp. 1-4.

Author



Yair Wiseman, He got a Summa Cum Laude M.Sc. and a PhD from Bar-Ilan University and completed two Post-Doc - one at the Hebrew University of Jerusalem and one in Georgia Institute of Technology.

Dr. Wiseman's research interests include Computational Transportation Science, Intelligent Transportation Systems, Process Scheduling, Hardware-Software Codesign, Memory Management, Computer Clusters, Data Compression, JPEG, Embedded Systems, Real-Time Systems and Operating Systems.

Dr. Wiseman is on the editorial board of several journals, a member of dozens of conference committees and a reviewer of many scholarly journals. Dr. Wiseman authored two books as well.

In addition, Dr. Wiseman has been teaching in many institutes including Bar-Ilan University, The Hebrew University of Jerusalem, Israel Aircraft Industry, Holon Institute of Technology and Jerusalem College of Technology.

Dr. Wiseman has been supervising many graduate students and an interesting point is that Albert Einstein is Dr. Wiseman's academic great-great-grandfather (*i.e.* the advisor of the advisor of the advisor of Dr. Wiseman's advisor).

Dr. Wiseman has collaborated with other partners and received research grants to run an active laboratory from inter alia Sun Microsystems, Intel, Polak Foundation and the Open University.

Dr. Wiseman is an international expert who has reviewed and evaluated several large projects of the European Union, Israel Science Foundation, MB Logic and more. Dr. Wiseman's papers have been published in many venues around the world.

