A Machine Learning Based Scheme for Double JPEG Compression Detection

Chunhua Chen, Yun Q. Shi, and Wei Su

New Jersey Institute of Technology, Newark, NJ, USA 07102
cc86@njit.edu, shi@njit.edu

Abstract

Double JPEG compression detection is of significance in digital forensics. We propose an effective machine learning based scheme to distinguish between double and single JPEG compressed images. Firstly, difference JPEG 2-D arrays, i.e., the difference between the magnitude of JPEG coefficient 2-D array of a given JPEG image and its shifted versions along various directions, are used to enhance double JPEG compression artifacts. Markov random process is then applied to modeling difference 2-D arrays so as to utilize the second-order statistics. In addition, a thresholding technique is used to reduce the size of the transition probability matrices, which characterize the Markov random processes. All elements of these matrices are collected as features for double JPEG compression detection. The support vector machine is employed as the classifier. Experiments have demonstrated that our proposed scheme has outperformed the prior arts.

1. Introduction

Double JPEG (Joint Photographic Experts Group) compression indicates that an image was originally a JPEG image and has been compressed once again by JPEG, which happens frequently nowadays. More specifically, we talk about cases in which the quality factor in the second JPEG compression is different from that in the first one.

Researchers focus their study on double JPEG compression for digital forensic purposes. Double JPEG compression tells a given image's history. For instance, to pretend that a tampered image is an authentic image, another JPEG compression is often applied. It is also known as a sensitive factor in steganalysis [1]. Some approaches to double JPEG compression detection have been reported in literature.

Popescu found that the histogram of a JPEG mode of a double JPEG compressed image contains some periodic artifacts, which could thus be utilized to distinguish between double and single JPEG compressed images [2]. The observation was also reported in [3] independently, and three methods for estimation of the primary quantization matrix from a double JPEG compressed image were presented.

Fu et al. [4] found that the distribution of the first digits of JPEG coefficients follows a generalized Benford's law. Some image forensic applications of this model, including the detection of double JPEG compressed images, were discussed, although no detailed result on this matter was reported.

We propose a novel and effective scheme based on machine learning framework to identify double JPEG compressed images in this paper. Features are formulated from the JPEG coefficient 2-D array of a given JPEG image. Specifically, difference JPEG 2-D arrays, i.e., the difference between the magnitudes of the JPEG coefficient 2-D array and its shifted versions along various directions, are used to extrude double JPEG compression artifacts. Markov random process is then applied to model these difference arrays so as to utilize the second-order statistics. In addition, a thresholding technique is used to greatly reduce the size of each transition probability matrix (TPM), which characterizes the Markov process. Elements of TPM's are collected as features for double JPEG compression detection. The support vector machine is employed as the classifier.

The rest of this paper is organized as follows. We briefly review Popescu's method in Section 2. Then our novel scheme is presented in Section 3. Section 4 gives experimental results over 4,005 images. Section 5 makes discussions. Conclusions are given in Section 6.

2. Popescu's method and its performance

Chapter 4 in [2] focuses on double JPEG compression detection. Assume that a JPEG mode has undergone double quantization, which is part of double JPEG compression operation. Denote the first quantization step by b and the second one by a. If a/b is not an integer, double quantization introduces

periodic artifacts to the JPEG mode's histogram. More specifically, the magnitude of DFT (discrete Fourier transform) of the JPEG mode's histogram has a specific peak pattern. Popescu proposed to detect the very existence of the peak pattern, which indicates double JPEG compression. He also argued that if a/b was an integer, single and double quantized JPEG modes would have the same histogram and hence it was impossible to distinguish between them.

To test his method, the author built an image database of 100 images. Each image was double JPEG compressed with several pairs of quality factors. Only the first 10 JPEG modes (in the zig-zag scanning order) of the luminance channel were employed to detect double JPEG compression artifacts. True positive rates, obtained by counting the correctly detected double JPEG compressed images using the thresholds deter-mined from the 100 single JPEG compressed images, are shown in Table 1, where the first quality factor is held constant on the rows while the second one is held constant on the columns. The "N/D" (not detectable) entries in Table 1 correspond to pairs of quality factors that yield double JPEG compressed images indistinguishable from single JPEG compressed images by his method.

3. Proposed approach

Popescu's method works quite well. However, it fails in the cases with "N/D" in Table 1. We introduce a novel scheme, in which, double JPEG compression detection is considered as a two-class pattern recognition task, i.e., a test JPEG image is to be classified as either single or double JPEG compressed.

3.1. Feature generation

In general, a color JPEG image has Y, C_b, and C_r components. For simplicity, in this paper, we only apply our feature generation scheme to the Y component. Please note that each component can be independently used to generate a subset of features following the same procedure, as shown in Figure 1.

3.1.1. JPEG 2-D array. For a given image, consider the 2-D array consisting of all the JPEG coefficients arranged block by block without overlapping, which is referred to as JPEG coefficient 2-D array. We only consider the magnitudes of these coefficients. The resultant 2-D array is called JPEG 2-D array. The reason to consider magnitude is twofold. Double JPEG compression seldom changes the sign of a JPEG coefficient. Furthermore, our proposed scheme utilizes the difference between an element in a JPEG 2-D array
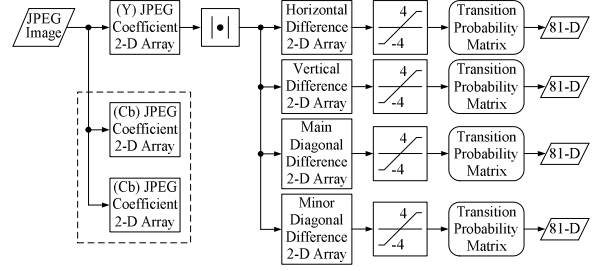

Figure 1. Feature generation procedure

Table 1. True positive rates given in [2] (%)

$Q_1 \backslash Q_2$	50	55	60	65	70	75	80	85	90	95
50	–	100	100	100	100	100	100	100	100	100
55	70	–	99	100	100	100	100	100	100	100
60	98	85	–	100	100	100	100	100	100	100
65	100	100	100	–	100	100	100	100	100	100
70	100	100	100	97	–	100	100	100	100	100
75	93	100	100	100	100	–	100	100	100	100
80	96	100	98	100	100	100	–	100	100	100
85	74	65	100	98	100	100	100	–	100	100
90	48	63	78	89	99	100	100	100	–	100
95	**N/D**	**N/D**	73	98	**N/D**	**N/D**	**N/D**	100	100	–

and its neighbors. Difference between two non-negative numbers is less likely to be truncated by the thresholding operation introduced later.

3.1.2. Difference JPEG 2-D array. It is expected that double JPEG compression artifacts can be enhanced by observing the difference between an element and its immediate neighbors in a JPEG 2-D array.

Denote the JPEG 2-D array generated from a given JPEG image by $F(u,v)$ ($u \in [0, S_h\text{-}1]$, $v \in [0, S_v\text{-}1]$), where S_h and S_v are the size of the JPEG 2-D array along the horizontal and vertical direction, respectively. We can generate four difference 2-D arrays (difference arrays in short) as follows:

$$F_h(u,v) = F(u,v) - F(u+1,v), \tag{1}$$

$$F_v(u,v) = F(u,v) - F(u,v+1), \tag{2}$$

$$F_d(u,v) = F(u,v) - F(u+1,v+1), \tag{3}$$

$$F_m(u,v) = F(u+1,v) - F(u,v+1), \tag{4}$$

where $F_h(u,v)$, $F_v(u,v)$, $F_d(u,v)$, and $F_m(u,v)$ denote difference arrays along the horizontal, vertical, main diagonal, and minor diagonal directions, respectively.

3.1.3. Transition probability matrix. We model the difference arrays defined above by using one-step Markov random process. According to the random process theory, a transition probability matrix (TPM) can be used to characterize a Markov process [5].

To further reduce computational cost, we resort to a thresholding technique. Based on our statistical study on a large natural image dataset, this threshold is selected as 4 in our work. If the value of an element in a difference array is either larger than 4 or smaller than -4, it will be represented by 4 or -4 correspondingly. Elements of the one-step TPM associated with the horizontal difference JPEG 2-D arrays are given by

$$p\{F_{h\to 1}=n|F_h=m\}=\frac{\sum_{v=0}^{S_v-2}\sum_{u=0}^{S_h-2}\delta\{F_h=m,F_{h\to 1}=n\}}{\sum_{v=0}^{S_v-2}\sum_{u=0}^{S_h-2}\delta\{F_h=m\}}, \quad (5)$$

where m, $n \in [-4, 4]$; $F_h\psi$ denotes $F_h(u^c\psi v)$, $F_{h\to 1}$ denotes $F_h(u\to 1^c\psi v)$, i.e., the element are shifted one step along the direction of different 2-D array formation; $\delta(A)=1$ if A holds or 0 otherwise. The elements of the one-step TPM's associated with the vertical, main diagonal, and minor diagonal difference JPEG 2-D arrays can be computed the similar way.

In a word, we have 9×9 elements for the TPM along each direction. As a result, a feature vector of 324 elements is obtained for each given JPEG image. All these elements are serving as features for double JPEG compression detection.

3.1.4. Effect of double JPEG compression on TPM.
As discussed in Section 5, the double JPEG compression leaves statistical artifacts among elements of the difference 2-D array, which are caused by the rounding errors in the double JPEG compression. The elements in a difference 2-D array of a single JPEG compressed image are often correlated. The double JPEG compression artifacts disturb the difference 2-D array, weakening the correlation among elements of the difference 2-D array of a double JPEG compressed image. Consequently, the associated TPM will have its high value zone spread from its center towards boundaries, leaving the double JPEG compression artifacts detectable by our proposed scheme.

3.2. Classification

The support vector machine (SVM) is used as the classifier in our proposed scheme, the Matlab code of which is downloaded from [6]. In this work, the polynomial kernel with degree two is used.

4. Experiments and results

We use 4,005 uncompressed images in our experimental works from three sources: 1,338 images of 384×512 from UCID [7], 1,124 images of 800×600 from Sun Yat-Sen University [8], and 1,543 images of 1,500×2,100 from NRCS Photo Gallery [9].

These images are first compressed to JPEG images using various quality factors (see Q_1 in Table 1), resulting in single JPEG compressed images. Then each single JPEG compressed image is recompressed using a different quality factor (see Q_2 in Table 1), resulting in a double JPEG compressed image.

We randomly select 5/6 of single JPEG compressed images with quality factor Q_2 and 5/6 of double JPEG compressed images with quality factor Q_1 followed by quality factor Q_2 to train the SVM classifier and the remaining 1/6 single and double JPEG compressed images to test the trained classifier. Detection accuracies are reported in Table 2. Note that these accuracies are averaged over 20 random experiments.

Table 2. Detection results: accuracies (%)

Q_1\Q_2	50	55	60	65	70	75	80	85	90	95
50	–	99	100	100	100	100	100	100	100	100
55	99	–	100	100	100	100	100	100	100	100
60	100	99	–	100	100	100	100	100	100	100
65	100	100	100	–	100	100	100	100	100	100
70	100	100	100	100	–	100	100	100	100	100
75	100	100	100	100	100	–	100	100	100	100
80	100	100	100	100	100	100	–	100	100	100
85	99	97	100	100	100	100	100	–	100	100
90	94	99	99	99	99	100	100	100	–	100
95	**81**	**82**	87	87	**92**	**91**	**97**	95	100	–

5. Discussions

We can see from Tables 1 and 2 that our proposed scheme outperforms Popescu's method in general. Especially, for all of five "N/D" cases in Table 1, our proposed scheme has achieved very high detection rate.

Theoretically, when a JPEG mode is quantized with a quantization step b followed by another quantization step a and if a/b is an integer, the JPEG coefficients of a JPEG mode are the same as that undergoing a single quantization with the quantization step a. As a result, single and double quantized JPEG modes have the same histogram and hence it's impossible to distinguish between them, and this is the reason why [2] fails on these "N/D" cases. However, in practice, double JPEG compression involves discrete cosine transform (DCT), quantization, dequantization, inverse DCT. In these processes, rounding errors are inevitable.

Comparing our proposed scheme with that in [2], we may have noticed the following. 1) [2] focuses on the histogram of each JPEG mode, which is of the first-order statistics. In contrast, our proposed scheme relies on the Markov process, and TPM applied to the difference JPEG 2-D arrays, which are of the second-order statistics. Furthermore, to effectively catch

artifacts left by double JPEG compression, our proposed scheme generates difference arrays along various directions. 2) [2] treats each JPEG mode independently. Besides, in the experiments generating Table 1, only the first 10 modes are considered. In contrast, our proposed scheme utilizes all the JPEG coefficients and hence all the JPEG modes.

But by no means has the above observation meant that Popescu's method will be necessarily more effective if more JPEG modes are considered. We have examined and compared the DFT magnitude spectrum of the histogram of JPEG modes 11, 12, 13, 14, and 15 in a single JPEG compressed image and that in a double JPEG compressed image in one "N/D" case (Q=55 vs. Q_1/Q_2=95/55). It is found that the single and double compressions are indistinguishable. In other words, even when the first 15 modes are considered, Popescu's method will still not be able to distinguish the Q=55 vs. Q_1/Q_2=95/55 case.

We have also conducted some experiments with our proposed scheme working on the JPEG 2-D array constructed from only the first 10 JPEG modes over the 4,005 images. When detecting the N/D case Q=70 vs. Q_1/Q_2=95/70, our proposed scheme can achieve the accuracy of 66.86%, while [2] fails (see Table 1). In another non-"N/D" case, Q=55 vs. Q_1/Q_2=70/55, our proposed scheme can achieve accuracy of 99.95%.

It is observed that, when a/b is an integer (e.g., the "N/D" cases in Table 1), the rounding errors do exist and leave artifacts with the double JPEG compressed images, which can be caught by our proposed scheme.

In addition to the five "N/D" cases, which can now be correctly detected, our method also improves the detection performance for some cases with high first quality factor and low second quality factor (see the bottom left portion of Table 1). For example, for the case Q=50 vs. Q_1/Q_2=90/50, Popescu's method can only identify 48% double compressed images, while our proposed scheme has achieved accuracy of 94%.

6. Conclusions and future work

In this paper, we have presented our newly established double JPEG compression detection scheme and demonstrated its effectiveness. We summarize our paper as follows:
1) Popescu's method, which is based on the "periodicity" pattern of histograms of the JPEG modes undergoing double JPEG compression, has achieved quite good performance in general. However, it fails for those "N/D" cases.
2) Rounding errors occurring in DCT, quantization, and inverse DCT during double JPEG compression

are inevitable, even when the quan-tization step in the first JPEG compression is a divisor of that in the second compression.
3) Difference JPEG 2-D arrays along different directions enhance rounding errors, the artifacts left with double JPEG compression.
4) Markov process and associated transition probability matrix efficiently catch the above mentioned statistical artifacts.
5) Thresholding strategy reduces computational cost.
6) In addition to feature construction, the proposed machine learning framework includes training and testing with the SVM, achieving promising performance in double JPEG compression detection.
7) Sometimes the second JPEG compression has inconsistent block splitting to the first compression, which is called shifted double (SD) JPEG compression. Since methods in [2] and [3] are based on individual JPEG modes, they are not expected to work effectively for the SD case. Instead, our proposed scheme utilizes the second-order statistics of the global JPEG coefficient 2-D array. Hence, when applying to the SD case, its effectiveness is anticipated. The performance may be degraded a little bit though.

References

[1] Y. Q. Shi, C. Chen, W. Chen, and M. P. Kaundiny, "Effect of recompression on attacking JPEG steganographic schemes ─ an experimental study", *ISCAS07*, New Orleans, LA, USA, May 27-30, 2007.
[2] A. C. Popescu, "Statistical tools for digital image forensics", PhD thesis, Dartmouth College, Hanover, NH, USA, December 2004 (advised by H. Farid). http://www.cs.dartmouth.edu/farid/publications/apthesis05.pdf.
[3] J. Lukas and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images", *Digital Forensic Research Workshop*, Cleveland, OH, USA, August 2003.
[4] D. Fu, Y. Q. Shi, and W. Su, "A generalized Benford's law for JPEG coefficients and its applications in image forensics", *SPIE Electronic Imaging*, San Jose, CA, USA, 2007.
[5] A. Leon-Garcia, *Probability and random processes for electrical engineering*, 2nd Edition, Addison-Wesley Publishing Company, 1994.
[6] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines", 2001. Available online at http://www.csie.ntu.edu.tw/~cjlin/libsvm.
[7] http://vision.cs.aston.ac.uk/datasets/ucid/ucid.html.
[8] http://ist.sysu.edu.cn/en/index_en.asp.
[9] http://photogallery.nrcs.usda.gov/.