ABSTRACT

In this paper, we propose a sector-wise JPEG fragment point classification approach to classify normal and erroneous JPEG data fragments with the minimum size of 512 Bytes per fragment. The contributions of this work are two-folds: 1) a sectorwise JPEG erroneous fragment classification approach is proposed 2) a new DCT coefficient analysis method is introduced for the JPEG image content analysis. Testing results on a variety of erroneous fragmented and normal JPEG files prove the strength of this operator for the purpose of forensics analysis, data recovery and abnormal fragment inconsistencies classification and detection. Furthermore, the results also show that the proposed DCT coefficient analysis method is efficient and practical in terms of classification accuracy. In our experiment, the proposed classifier yields a FP rate of 4.89% and a TP rate of 96.6% in terms of erroneous JPEG fragment detection.

Index Terms—DCT coefficient analysis; Erroneous fragment classification; JPEG carving; Digital forensics

1. INTRODUCTION

With the fast progress of electronic technology, enormous digital records such as images, videos, documents, audios are generated and stored on digital storage media. For images, JPEG is adopted by most image capture/storage devices and therefore, the JPEG image is one of the most important sources of evidence during forensic evidence analysis. However, when files are stored on digital media, they are often fragmented. The description of the logical structure of a file and its underlying raw data fragments is typically stored separately from the file data. If such a description is lost, the operating system would not be able to read the file data from the storage device, even when all its corresponding data fragments still reside in the device. This can occur, for example, when the file is deleted by the operating system but the underlying data storage has not been recycled to store other information, or when the file system itself is damaged. Scenarios like these are of great interests in forensics applications, or the recovery of files that are either accidentally deleted or lost due to faulty file systems.

A conventional approach to JPEG fragmentation point detection is to parse the fragment data by joining it with the previously decoded data using a standard JPEG decoder, and expect that the erroneous fragment would cause a warning from the decoder. However, this approach is not reliable since there is a good chance that a fragment, which does not belong to the JPEG image, can pass through the JPEG decoder without triggering any warning. In fact, as shown in [1] [2], a faulty JPEG image can be fully validated and decoded by a standard JPEG decoder for various cases.

A common observation is that a decoded erroneous fragment would have a different and inconsistent appearance with the previously decoded normal JPEG content [1], as shown in Fig. 3 to 6. Thus, a more reliable erroneous fragment detection approach can be achieved by using the inconsistent appearances of the decoded, incorrectly joined fragment. In this paper, a sector-wise JPEG fragment classification approach is proposed to classify the normal and erroneous JPEG fragments, and to detect the start point/sector of the erroneous fragment. We assume the JPEG header is not recycled or damaged and can be found by searching for the “hex: FFD8” as the JPEG SOI marker [3][5][4]. Our objective is, given a damaged JPEG file with some normal JPEG fragments which can be correctly decoded, to determine if the fragment that follows those fragments is a correct fragment that belong to this JPEG file or not (i.e. erroneous fragment).

To achieve this goal, we extract the fragment inconsistency features from the content of erroneous fragment with the minimum size of 512 bytes, by applying our proposed JPEG coefficient analysis algorithms. The final classifier is trained with a linear support vector machine (SVM), and the evaluation for performance of the proposed classifier is conducted based on our experiment. Compared with the existing work [1], our proposed classifier is significantly stronger and practical in terms of classification accuracy.

In the rest of this paper, we introduce our feature extraction algorithms including a Horizontal Versus Vertical (HVV) descriptor, and an Intra- and Inter-sector Difference Evaluator...
(IIE) in Section 2. The experiment conducted to evaluate our JPEG erroneous fragment classification approach and results are presented in Section 3. The introduction to future work and conclusion are given in Section 4 and 5. To simplify the description, we would use the term “JPEG fragment” for the normal JPEG fragment, and the term “erroneous fragment” for the erroneous fragment joined to a corresponding image data, resulting in a corrupted image.

2. DESIGN OF THE ERRONEOUS FRAGMENT DETECTION APPROACH

The purpose of this work is to classify the normal and erroneous JPEG data fragments and detect the starting point of the erroneous fragment. The content inconsistencies of the erroneous fragment are extracted based on the JPEG coefficient analysis. We found out that, the variance of this content inconsistency is considerably large across different corrupted JPEG images. As shown in Fig. 6 to 14, the classification capability of a single feature is limited. One content description feature would be insufficient to detect the erroneous fragment from the corrupted JPEG image. Thus, we extract a group of features from different content aspects to obtain more information regarding the content inconsistencies. A linear SVM is used for the training and classification. The rest of this section is organized as follow: Section 2.1 introduces a new JPEG DCT component Horizontal Versus Vertical (HVV) descriptor which describes the emphasized direction of each given DCT block without performing JPEG decoding, Section 2.2 introduces a new Intra- and Inter-sector Difference Evaluator (IIE) which extracts the image content changes by using both the intra-sector differences and inter-sector differences, and Section 2.3 introduces the features derived from our proposed descriptor and methods.

2.1. A Novel Horizontal Versus Vertical (HVV) Descriptor for DCT Coefficients Analysis

The JPEG encoding process uses the discrete cosine transform (DCT) which encodes a sequence of data, such as the intensity values of 64 pixels on an 8x8 block, into their corresponding DCT coefficients. These encoded JPEG coefficients are the summation results of the cosine functions oscillating at different frequencies. For the JPEG coefficient analysis, most of the existing methods use the correlation of the DCT coefficients at the upper-left and lower-right triangle regions to extract the degree of emphasis on the lower frequency and higher frequency components for the given 8x8 DCT block.

In practical applications, there is a major weakness of these DCT block frequency analysis methods. For some of the JPEG coefficient blocks, the higher frequency coefficients which are located within the lower-right triangle region are either all 0 or have values for very few elements. Such cases usually occur for the 8x8 image blocks which are covering the scenes with little high frequency contents, such as the blue sky. The high quantization steps used during the JPEG encoding can also result in such situations. For these DCT blocks which have insufficient information to be analyzed within the high frequency region, the performance of those JPEG coefficient analysis methods which depend on such information would be unreliable and unstable.

Where 0 ≤ u, v < 8 and

\[
\alpha(x) = \begin{cases} 
\sqrt{2} & x = 0 \\
1 & x > 0
\end{cases}
\]

The design of our novel DCT coefficient HVV descriptor has a focus on extracting the direction emphasis of the vertical and horizontal appearance, for the given 8x8 DCT coefficients block. As the calculation for the DCT shown in Equation (1), for each coefficient \( G_{u,v} \), \( u \) is the horizontal spatial frequency, and \( v \) is the vertical spatial frequency, for the integers 0 ≤ u, v < 8. The differences among \( u, v \) for 64 coefficients would lead to different emphasis on the directional appearances for the decoded JPEG image blocks, as shown in Fig. 1.

Thus, we gather all the DCT coefficients, within the upper-right triangle shown in Fig. 2. These coefficients have an emphasis on the vertical direction. We also gather all the DCT coefficients within the lower-left triangle, which have emphasis on the horizontal direction. Finally, we form our descriptor as illustrated in the Equation (2) and (3).
$$D_{HVV} = S_V - S_H$$

$$S_V = \sum_{i=0}^{i<7} \sum_{j=i+1}^{j<8} |C_{i,j}| \quad \text{and} \quad S_H = \sum_{j=0}^{j<7} \sum_{i=j+1}^{i<8} |C_{i,j}|$$

$C_{i,j}$ indicates the corresponding coefficient element in the (i, j) position within the 8x8 coefficient block. With our HVV descriptor, an image blob which has a focus on the vertical appearance would yield a positive value, while a block with a horizontal appearance would yield a negative value. Moreover, from our experiments, the degree of emphasis on these directions of appearances can be reflected by the magnitudes of the HVV values.

One main advantage of our novel HVV descriptor is the stable performance it achieves. Since it utilizes the upper-right and lower-left regions of the DCT coefficient block, even if the higher frequency part which is the lower-right triangle region of the coefficients provides insufficient information, such as the previously mentioned all-0 situation, our operator can still perform well. This is due to the definite presence of the directional information in natural photo images, which is an important quality in the applications of forensics investigation, common users data recovery and JPEG image anomalies detection. This novel descriptor is therefore more effective and thus more practical for the applications in realistic JPEG coefficient analysis.

To illustrate the proposed algorithm, an image region (Fig. 3 left diagram) from a corrupted image is extracted from our JPEG dataset. We provide a zoomed-in version (Fig. 3 right diagram) beside it, showing the 3x3 block in the image, with each of nine gridded blocks referring to an 8x8 pixels blob.

There are three rows in the zoomed-in region, and each row contains 3 zoomed-in blocks. The first row belongs to the normal fragment, and the second and third rows are from the corrupted region. An obvious observation is that some of the blocks emphasize clearly on the horizontal direction, such as those three blocks in the second row, and some blocks emphasize clearly on the vertical direction, such as the middle and right block in the third row, while the rest of the 9 blocks have blurry directional emphasises. The corresponding values obtained from our HVV descriptor for the 9 blocks on the luminance channel are illustrated in Table 1. It is clearly shown that, the magnitudes of the HVV values indicate the degree of vertical and horizontal emphasis.

2.2. Design of An Intra-sector and Inter-sector Difference Evaluator (IIE)

Another focus of this work is to derive and differentiate between the inconsistencies of the incorrectly joined fragments (i.e. a JPEG image fragment joined with an erroneous fragment belonging to another file) in a corrupted JPEG image. Based on our experimental results, the inter-sector vertical difference and the intra-sector horizontal difference are both feasible features for classifying the erroneous and normal fragments. Here, the inter-sector vertical difference refers to the magnitude difference of the JPEG blocks which are belonging to different storage sectors but are vertically connected. The intra-sector horizontal difference refers to the magnitude difference of the JPEG blocks which are belonging to the same storage sector, and are horizontally connected. For both features, the variances of the normal JPEG fragments and the erroneous fragments are considerably large. Therefore, to increase the stability for handling a vast range of corrupted and non-corrupted JPEG image, we introduce another operator to identify the inconsistencies from both the intra and inter-sector differences.

During decoding, the JPEG data can only be correctly interpreted with the corresponding decoding information, such as the quantization table in the header for this JPEG file. When using the decoding information to process an erroneous fragment, which can be of any file type, the intra-sector consistency of the appearance in the decoded content would be affected and the image would have an unnatural look, as shown in the rectangular region B in Fig. 4. In Fig. 4, the
Fig. 4. An example of the erroneous fragment

Fig. 5. An example illustrating the inefficiency of inter-sector difference feature

zoomed-in version of the region is shown below its original image. However, for images with a relatively complex scene, the content of a true JPEG region can be very inconsistent too. One example case is shown in the rectangular region A of Fig. 4. Obviously, the intra-sector difference will not perform well on regions as shown in region A of Fig. 4, but an inter-sector difference such as the vertical DC component difference may work well. However, the inter-sector difference is not universally reliable either. The reason is that, for some cases, the difference between the connected JPEG and the erroneous fragment can be small, an example case is shown in Fig. 5.

Thus, we design our IIE operator, as illustrated in Equation (4) and (5), so that the intra and inter-sector differences can be combined to form a more efficient and reliable feature.

\[
Hd_{i,j} = V_{i,j} - V_{i,j-1} \quad \text{if } j > 0 \tag{4}
\]

\[
D_{IIE} = 1/n \sum_{j=1}^{n} |Hd_{i,j} - Hd_{i-1,j}| \quad \text{if } i > 0 \tag{5}
\]

\(Hd_{i,j}\) represents the horizontal intra-sector difference of two connected blocks. \(V_{i,j}\) can be one of the features such as the DC component or the HVV value for describing the block. The absolute value of the inter-sector difference is calculated based on the horizontal intra-sector differences for the vertically connected JPEG and the erroneous fragment blocks.

The average value of the vertical differences in the connected blocks within the read-in sector is calculated as one of our utilized features.

2.3. Utilized Features in Our System

We apply seven groups of features in our classification approach. They are 1) the result of using our IIE methods on the DC coefficients; 2) the standard deviation of the HVV values for the blocks of each sector 3) the result of using our IIE methods on the HVV values; 4) the intra-sector horizontal difference, which is the average intra-sector horizontal difference for the blocks that are horizontally connected with other blocks from the same sector; 5) the inter-sector vertical difference which is the average inter-sector vertical differences for the blocks that are vertically connected with other sector blocks; 6) the higher frequency analysis values and 7) the edge density.

There are three feature values for each of the first six feature groups. The three values are obtained from one luminance channel (Y) and two chrominance channels (CB and CR). These three channels describe the colour of the image content in three different directions. Including the descriptions for all three components would make the classifier more reliable, as more non-redundant distinguishing information is used. Therefore, our feature vector composes of these 18 feature values together with the last edge density value.

The standard deviation (SD) is used in the second feature group, to evaluate the intra-sector variance based on the HVV values. As shown in Fig. 3 to 5, the emphasis on the directions of the blocks from erroneous fragments are unnatural and the direction emphasis switches fast and frequently. Thus, the first sector of the erroneous fragment for each corrupted image usually yields a much higher SD value than the true JPEG sectors.

The ROC curves to illustrate the nine features of the 1st, 2nd, and 3rd groups are given in Fig. 6 to 14, respectively. The detailed introduction and illustrations to the rest of the 10 features within the last four feature groups (i.e. 4th, 5th, 6th, and 7th feature groups), are provided in our previous work, Section 3.3 and 3.4 in [1].

3. EXPERIMENTS

To evaluate our proposed approach, we implemented it on top of the “libjpeg” library, and randomly generated the corrupted JPEG files using a database that contains over 1200 publicly available photos of natural scenes (http://www.pdphoto.org). For generating each corrupted JPEG file, we divided a JPEG image which was randomly selected from our 1200 image set, into blocks of 512 bytes (logically), and randomly selected an erroneous fragmentation point that was beyond its first SOS marker [1]. This step ensures that the erroneous fragmentations occur in the compressed data stream but not in
the “header” part. Next, we randomly selected another (512-byte) sector-size of data from all other files, including different formats of documents, videos, audios, pictures etc., and appended the selected data to the JPEG file at the randomly selected erroneous fragmentation point to create a corrupted JPEG image.

The resulting images were then parsed by the standard “libjpeg” decoder, and it was only accepted into our data set of corrupted images when there was no error or warning generated by “libjpeg” (except the one common warning that indicated premature ending of the JPEG image). In other words, the experiment was designed to deal with corruptions that cannot be detected by a standard decoder.

We generated 3093 corrupted photos for use as our data set of images. As each of the corrupted images would have one erroneous fragment which is erroneously joined with the normal JPEG fragments, a vector of 19 features can be extracted from each of these 3093 images. From these 3093 erroneous fragment feature vectors, we randomly selected 2000 for our training data and 1000 for our testing data. We also extracted 30000 normal JPEG feature vectors from the randomly selected sectors in the randomly selected images from our non-corrupted JPEG image set. These 30000 feature vectors were randomly divided to 20000 feature vectors for training and 10000 feature vectors for testing use.

A linear SVM was used to train our classifier. We tested the classifier using our testing data. For the 1000 erroneous testing cases, the true positive rate (TP) was obtained at 96.6% and for the 10000 normal JPEG testing case, our approach yielded a TP rate of 95.11%. Therefore, in terms of the erroneous fragment detection, a false positive (FP) of 4.89% and a false negative (FN) rate of 3.4%, were achieved.
4. FUTURE WORK

It is important to note that our approach is designed to work as a fundamental/low-level classifier for any erroneous fragmented data and for all kinds of JPEG images encoded in the sequential mode. In our case, only very few sectors (usually 2 or 3), from the image are needed. Our method compares contents of each read-in sector with the image contents from a few sectors of normal JPEG data before it. This design makes our classifier work well and stably for most kinds of corrupted JPEG images. An even better performance of the fragment classification approach may be achieved by analyzing on the following consequential content after the target sector. As the minimum size for the erroneous fragment of the JPEG image is usually greater than 512 bytes for most storage systems (i.e. cluster size is usually a multiple of more than 1 of sector size), the classification can be improved by examining if a detected erroneous fragment sector is followed by another erroneous fragment sector, if so, the confidence of the classification decision can be increased, otherwise, the confidence can be decreased.

The second possible way to improve the current design is to improve the efficiency of the feature utilization. We use 19 features in this research. The efficiencies among those features vary. Thus, it is possible to eliminate some of the less efficient features, so as to achieve better computation efficiency during the classification process, without sacrificing the classification accuracy very much. This possible improvement is especially favourable for the application which needs a fast processing speed or deals with a huge amount of target data.

The current design of the classification approach can only work for the JPEG image encoded in the sequential mode. Fu-
ture work is needed to perform the erroneous fragmented data classification on the JPEG encoded with other modes such as the progressive mode.

5. CONCLUSION

In this paper, we propose a new classification approach to classify the erroneous and normal fragments for JPEG images. Our method is based on processing each read-in sector of 512 bytes with using the DCT coefficient analysis methods to extract the inconsistencies features.

The merits of this work can be weighed through the two following original contributions:
1) A practical sector-wise JPEG erroneous fragment detector which supports sector-wise JPEG data processing is proposed. Compared to the existing work of [1], our detection approach is capable of realistic applications because of the significant improvement on the detection performances on realistic data. Our detector yields a FP rate of 4.89% and a FN rate of 3.4% from our experiment involving 10000 correct JPEG test cases and 1000 JPEG test cases with erroneous fragments.
2) A novel DCT coefficients analysis descriptor “Horizontal Versus Vertical (HVV) Descriptor” is introduced. This approach is significantly different from the commonly used JPEG image DCT analysis approach. This method does not simply perform the analysis on the frequency distributions but involves a novel approach of identifying and analyzing the DCT coefficients with an emphasis on the horizontal and vertical components. As shown in Section 2.1, this new method is highly beneficial in JPEG carving and holds promising potential in various other applications requiring JPEG coefficient analysis. To the best of our knowledge, our proposed descriptor is the first work utilizing the magnitude difference of the horizontal-focus and the vertical-focus coefficients for the directional analysis on the JPEG coefficient blob.

6. REFERENCES


