Image Forgery Detection Based on Statistical Features of Block DCT Coefficients

Moussa EL OUAFI
Department of Mathematics
École Normale Supérieure Paris-Saclay - Paris-Saclay University
moussa.el_ouafi@ens-paris-saclay.fr
Original paper : Link

Abstract

A significant number of currently exploited algorithms can either detect splicing image forgery or copy-move image forgery, but not both simultaneously. In this paper, the authors propose a unified approach to handle these two types of forgeries in the same instance. The new technique utilizes the artifacts resulting from manipulations performed on JPEG-encoded images. To understand the underlying principle of this new method, we first need to look at how JPEG compression works: an image is divided into disjoint blocks of size 8×8*, and then the discrete cosine transform (DCT) coefficients are evaluated for each block independently. When a compressed JPEG image has been tampered with, the statistical properties of the AC components of the block-DCT coefficients are modified. To capture this change, we propose using the standard deviation and the count of non-zero DCT coefficients corresponding to each of the AC frequency components independently. The images are cropped by removing a few rows and columns from the top left corner, and then the suggested features are evaluated for both the test image and its cropped version. The extracted feature vector is used in conjunction with a support vector machine (SVM) for the binary classification of images (authentic vs. forged). The experiments are conducted on a standard dataset (CASIA v1.0) of preprocessed forged images to support the theoretical principle of the proposed technique.*

1. Introduction

Thanks to the technological advancements that the world has seen in the last two decades, it has become possible, even for non-specialists, to have access to the means and knowledge that allow them to easily edit digital data, including images. This raises concerns regarding the credibility of digital documents, especially in an era when they are being used as official supporting documents, whether in court, at a hospital, or in a financial institution. Hence, it has become necessary to determine the integrity of digital images, which is now part of the responsibilities of digital image forensics. These professionals deal with verifying the authenticity of images that could have undergone one of the two most common alterations: splicing or copy-move manipulations. Splicing forgery consists of combining two or more images, whereas copy-move forgery involves copying a part of an image and pasting it onto the same image.

JPEG is the most common digital image format. In order to be compressed, a JPEG image is divided into disjoint blocks of size 8×8 pixels. For each block, the Discrete Cosine Transform (DCT) is evaluated and quantized using a standard quantization matrix. This makes the block discrete cosine transform (BDCT) the core step of JPEG compression, which means that any tampering with the image will create disturbances in the local statistics of BDCT coefficients. This change can be detected and used as an indicator of image forgery. Shi et al. [7] proposed an image splicing detection algorithm based on a natural image model containing moments of the characteristic function of wavelet sub-bands and Markov transition probabilities of difference 2-D arrays. These 2-D arrays were generated by applying a multi-size BDCT on a given test image. An additional splicing detection technique was proposed by He et al. [5]. They captured the inter-block and intra-block correlation between BDCT coefficients by expanding the original Markov features generated from the transition probability matrices to the DCT domain. Furthermore, three types of dependencies among the wavelet coefficients across positions, scales, and orientations were characterized by constructing more features in the DCT domain. In order to reduce the computational cost, the feature selection method SVM-RFE was used.

The second type of common image forgery is copy-move forgery, which consists of copying a patch of an image and pasting it somewhere else on the same image. Despite being common, it is difficult to detect copy-move alterations with the naked eye. Fridrich et al. [4] proposed the

first technique to detect copy-move forgery in which DCT coefficient-based features are extracted from small overlapping image blocks. The tampered areas were detected by performing similarity checks between lexicographically sorted feature vectors. However, this approach led to a number of false matches when applied to images containing large identical textured regions. The authors in [2] proposed a similar approach, in which they also extracted DCT coefficients as features for different block sizes. However, these techniques suffered from high computational complexity and improper detection of tampered areas when post-processing operations were applied to the tampered images.

All the techniques discussed so far work relatively well for either type of forgery. However, very few techniques reported in the literature are adapted for both types of forgery. One of the integrated techniques proposed by Alahmadi et al. [1] can perform copy-move and splicing forgery detection in the same instance. The method was based on the local binary pattern (LBP) and DCT. They transformed the LBP code of each block of the forged image into the DCT domain and then evaluated the standard deviation of these DCT block coefficients. However, the technique was not evaluated for various post-processing operations. Recently, Prakash et al. [6] proposed another integrated technique for detecting both splicing and copy-move forgery. An enhanced threshold method based on the Markov random process to extract the features from different color spaces was suggested in their study. However, they have not evaluated the scheme with a combined collection of authentic, spliced, and copy-move forged images for both datasets.

In this paper, a new integrated image forgery detection technique is presented. The core principle is to make use of the variation in statistical properties of AC coefficients of the entire image by computing the standard deviation and count of non-zero DCT coefficients corresponding to each AC frequency component independently. The suggested features are evaluated for the test image and its cropped version. The extracted feature vector is then used with an SVM classifier for identifying modified/unmodified images. The proposed scheme is tested with various pre-processed forged images available in the CASIA dataset [3], as showcased in the results section. In Section 3.3, two machine learning models using features provided by the main technique of this paper are compared. The rest of the paper is organized as follows: Section 2 gives a detailed description of the proposed forgery detection algorithm and the core mathematical ideas behind it. In Section 3, simulation results of the proposed approach are presented. Finally, Section 4 concludes the paper.

2. Proposed forgery detection algorithm

Fig. 1 below shows the structural outline of the proposed forensic detector:

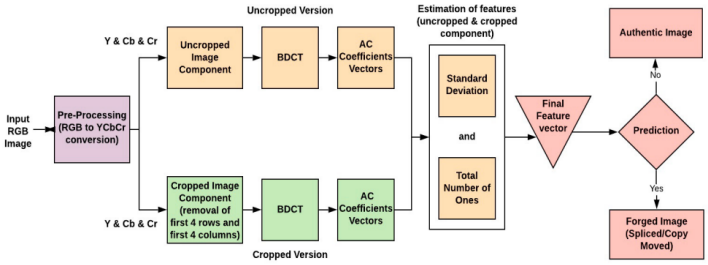


Figure 1. Scheme of the working principle of the proposed forgery detection algorithm.

2.1. Pre-processing

In this paper, the test (or original) image undergoes a pre-processing phase. First, it is converted into the YCbCr color space (where Y denotes the luminance component, and Cb and Cr represent the chrominance channels). In the YCbCr color space, most of the image information is contained in the Y channel. Moreover, most tampering traces are hidden in the chrominance channels. Since the human eye is more sensitive to the luminance component compared to the chrominance component, it becomes very difficult to detect tampering with the naked eye. For this reason, in this paper, all three channels are used to create and extract the final features.

A systematic flow of the proposed forgery detection method is presented on the next page in Algorithm 1.

2.2. Modeling the tampering traces in the DCT domain

Let Q be a 2-D matrix representing the color sub-image (either Y, Cb, or Cr) of the input image. The sub-image of size $J \times K$ is divided into N non-overlapping blocks of size 8×8 pixels, and then a 2D-DCT operation is performed on each block. In this case, each DCT block consists of one DC coefficient and 63 AC coefficients. In this paper, the focus is only on the behavior of AC coefficients. For clarity, the AC coefficients of each block are arranged into a vector:

$$A_i = [a_{2,i}, ..., a_{64,i}]^T, \quad 1 \leq i \leq N \tag{1}$$

A_i represents the vector of length 63, and $a_{k,i}$, $1 \leq k \leq 63$, is the respective AC coefficient in block i. The main aim is to analyze the behavior of the full input image, so all frequencies must be considered separately for all the N blocks.

Algorithm 1 Classification of Authentic vs Forged Images

Input: Potentially tampered image.
Output: Classification result (Authentic or Forged).

procedure :
Convert the RGB image (input image) into YCbCr color space.

for *each color sub-space (Y, Cb, Cr)* **do**
Divide the sub-image into N disjoint blocks of size 8×8.
Apply DCT to each block.
for *each block* **do**
Remove the DC coefficient.
Arrange the remaining 63 AC coefficients of the block into a column vector:
$A_i = [a_{2,i}, ..., a_{64,i}]^T, 1 \leq i \leq N$.
end
Concatenate column vectors of all N blocks to form a matrix of size $63 \times N$:
$M_A = [A_1, ..., A_N]$.
Compute the standard deviation over rows s_k and the number of ones a_k, $2 \leq k \leq 64$ of matrix M_A for each frequency component.
Form a vector of standard deviation: $S = [s_2, ..., s_{64}]^T$ and ones: $O = [O_2, ..., O_{64}]^T$.
Crop the sub-image by removing four rows horizontally and four columns from the top left corner.
Form the vector of standard deviation S_C and ones O_C for the cropped sub-image.
Concatenate the four vectors to obtain the feature vector of the sub-image:
$F_{sub-image} = [S^T, O^T, S_C^T, O_C^T]$.
end
Combine each of the sub-image feature vectors to form the final feature vector of the test image:
$F_V = [F_Y, F_{Cb}, F_{Cr}]$.
Apply feature vector F_V to the SVM classifier for detection of Authentic/Forged Image.

By concatenating the AC coefficients of each column vector, we construct the matrix below:

$$M_A = \begin{bmatrix} a_{2,1} & \dots & a_{2,N} \\ \vdots & \ddots & \vdots \\ a_{64,1} & \dots & a_{64,N} \end{bmatrix} \tag{2}$$

In Equation (2), the scalar located in row i and column j represents the AC coefficient for frequency number i and block number j of the sub-image. In this case, the row-wise standard deviation s_k is computed and arranged as a column vector, as shown below:

$$S = [s_2, ..., s_{64}]^T \tag{3}$$

where $s_k = 1.4826 \times \text{std}(a_{k,1}, ..., a_{k,N})$ and std is the standard deviation of a tuple of real numbers.

Similarly, the number of ones for each row (denoted o_k) is calculated and used to form a column vector:

$$O = [o_2, ..., o_{64}]^T \tag{4}$$

where $o_k = \sum_{1 \leq i \leq N} (a_{k,i} == 1)$ is the number of AC coefficients of a given frequency that are equal to 1, computed for all the blocks.

The next step involves cropping the sub-image by removing the first four rows and four columns from the top left corner. Then, all the steps described above for the non-cropped version are repeated to compute the standard deviation vector S_C and the ones vector O_C for the cropped sub-image. The feature vector of the sub-image is then derived as:

$$F_{sub-image} = [S^T, O^T, S_C^T, O_C^T] \tag{5}$$

To obtain the final feature vector F_V of the input image, each feature vector obtained from the three sub-images (Y, Cb, and Cr) is concatenated as shown below:

$$F_V = [F_Y, F_{Cb}, F_{Cr}] \tag{6}$$

The input image may be authentic or fake (either spliced or copy-move forged). Fig. 2(a, b, c) shows examples of an authentic image, a copy-move image, and a spliced image, respectively.

We consider the three matrices F_V^A, F_V^C, and F_V^S that represent the feature vectors of the authentic image and the two forged images (shown in Fig. 2). Then:

$$D_V^{AC} = F_V^A - F_V^C \quad \text{and} \quad D_V^{AS} = F_V^A - F_V^S \tag{7}$$

where D_V^{AC} is the feature difference between the authentic image and the copy-move forged image, and D_V^{AS} is the feature difference between the authentic image and the spliced image.

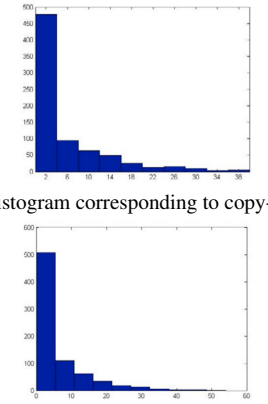
(a) Authentic image

(b) Tampered image: Copy-move

(c) Tampered image: Splicing

Figure 2. Authentic and forged images.

The histograms of both of these difference vectors are presented in Fig. 3. It can be observed that there is a noticeable difference in the features in both cases. Hence, the proposed feature vector extracted from the test image is discriminative enough and can be exploited for image forgery detection.



(a) Histogram corresponding to copy-move

(b) Histogram corresponding to splicing

Figure 3. Histograms of the feature difference vectors.

2.3. OCSVM: One-Class Support Vector Machine

One-class support vector machines (OCSVM) are a machine learning algorithm that aims to estimate the support of a distribution.

Given a dataset $\{y_1, \cdots, y_i, \cdots, y_N\}$, where $y_i \in \mathbb{R}^d$, the basic idea behind the OCSVM is to find a hyperplane defined in a high-dimensional Hilbert feature space $\mathcal{F}$ with maximum margin separation from the origin. The data are mapped to the space $\mathcal{F}$ through a nonlinear transformation $\phi(.)$. Then, the problem of separating the dataset from the origin is equivalent to solving the following quadratic program:

$$\underset{w,a,\xi,\rho}{\text{Minimize}} \ \frac{1}{2}\|w\|^2 + \frac{1}{vN}\sum_{i=1}^{N}\xi_i - \rho$$

$$\text{Subject to } w \cdot \phi(y_i) \geq \rho - \xi_i, \quad \xi_i \geq 0, \quad \forall i = 1, \cdots, N$$

where w is a vector perpendicular to the hyperplane in $\mathcal{F}$, ρ is the distance to the origin, $\xi_i \geq 0$ are slack variables to deal with outliers (forged samples) that may be included in the training data distribution, and $v \in (0, 1]$ is the parameter to control the trade-off between the number of examples of the training set mapped as positive by the decision function:

$$f(y) = \text{sign}((w, \phi(y)) - \rho).$$

Let $k(y_i, y_j) = \exp\left(-\frac{\|y_i - y_j\|^2}{2\sigma^2}\right)$ be the reproducing kernel (r.k.) of $\mathcal{F}$. (k is often called the RBF or Gaussian Kernel.)
Here, $\sigma > 0$ stands for the kernel width parameter. In the feature space $\mathcal{F}$, the distance between two mapped samples y_i and y_j is:

$$\begin{aligned}\|\phi(y_i) - \phi(y_j)\|^2 &= k(y_i, y_i) + k(y_j, y_j) - 2k(y_i, y_j) \\ &= 2\left[1 - \exp\left(-\frac{\|y_i - y_j\|^2}{2\sigma^2}\right)\right]\end{aligned}$$

Using the Lagrangian method, we get the following dual optimization problem:

$$\alpha^* = \underset{\alpha}{\text{Argmin}} \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j k(y_i, y_j)$$

$$\text{Subject to } \sum_{i=1}^{N}\alpha_i = 1, \quad 0 \leq \alpha_i \leq \frac{1}{vN}, \quad \forall i = 1, \cdots, N$$

Samples y_i that correspond to $0 < \alpha_i^* < \frac{1}{vN}$ are called *support vectors* (according to the complementary slackness condition in KKT).

Let N_{SV} stand for the number of support vectors; then the discriminant function is reduced to:

$$f(y) = \text{sign}\left(\sum_{i=1}^{N_{SV}}\alpha_i^* k(y, y_i) - \rho\right)$$

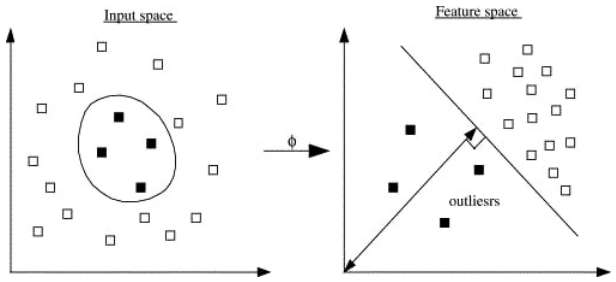


Figure 4. Illustration of the working principle of the OCSVM kernel trick.

2.4. Image classification

Image forgery detection is a binary classification problem. In the scheme of this paper, a support vector machine (SVM) is used as the classifier because of its promising results across a multitude of applications. For the sake of comparison, we also use two more classifiers: Random Forest and XGBoost.

3. Simulation

3.1. Image dataset

The approach, carried out conjointly with the three classifiers (SVM, which was used in the original paper, and two other classifiers: Histogram Gradient Boosting Classifier (HGBC) and XGBoost), is tested on a tampered image dataset (CASIA V1.0), which was made available by the Institute of Automation Chinese Academy of Science and is more fitting for real-world data. The dataset contains 800 authentic JPEG images and 921 tampered JPEG images of size 384×256.

3.2. Metrics

In this section, we compare three approaches using either a Support Vector Machine, XGBoost, or Histogram Gradient Boosting Classifier (HGBC). The chosen performance metrics are the ROC curve, Area Under Curve Precision-Recall (AUC-PR), Accuracy, Precision, Recall, and F-Score.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}, \quad \text{Recall} = \frac{TP}{TP+FN}$$

$$\text{Precision} = \frac{TP}{TP+FP}, \quad \text{F-Score} = 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$$

The results are presented in Fig. 5, Fig. 6, and Fig. 7 below:

SVM Metrics:

	precision	recall	f1-score	support
0	0.79	0.83	0.81	149
1	0.87	0.83	0.85	196
accuracy			0.83	345
macro avg	0.83	0.83	0.83	345
weighted avg	0.83	0.83	0.83	345

XGBoost metrics:

	precision	recall	f1-score	support
0	0.82	0.87	0.84	149
1	0.89	0.85	0.87	196
accuracy			0.86	345
macro avg	0.85	0.86	0.86	345
weighted avg	0.86	0.86	0.86	345

HGBC metrics:

	precision	recall	f1-score	support
0	0.78	0.89	0.83	149
1	0.91	0.81	0.86	196
accuracy			0.85	345
macro avg	0.85	0.85	0.85	345
weighted avg	0.85	0.85	0.85	345

Figure 5. Metrics comparison of the three models.

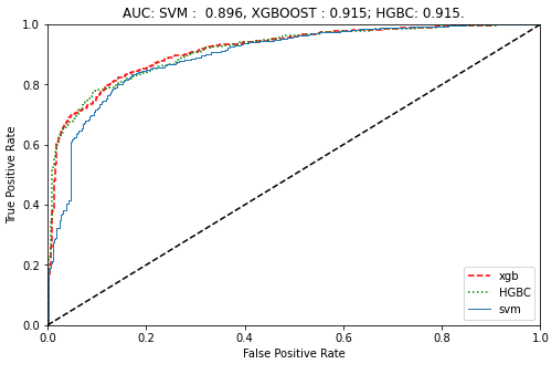


Figure 6. ROC of the three models.

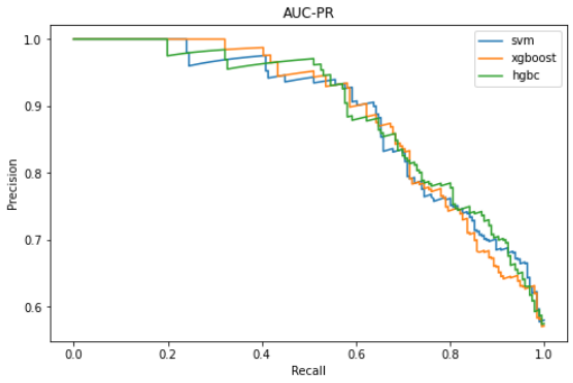


Figure 7. AUC-PR curve for optimal SVM, XGBoost, and HGBC.

The AUC of the SVM is computed to be 0.89, whereas the AUC of the XGBoost is 0.91, indicating that the latter slightly outperforms the former.

4. Conclusion

In the scope of this paper, we've reviewed and retested a new unified technique for image forgery detection (for copy-paste and splicing alterations), and we discussed the core idea behind it before moving on to the mathematical formulation that allows us to compute the feature vector, which is fed to the SVM. Finally, we've compared the SVM to two new models (HGBC and XGBoost) by feeding them the feature vector and comparing their performances using common metrics such as AUC and ROC. According to this experiment, we slightly improved the results of this technique by using HGBC instead of SVM. This work does not specifically address identifying which patches of the images were forged, which could be a valuable step toward creating a more robust model for image forgery detection. Indeed, it might be more efficient to first classify forged images before using other models to detect which regions have been forged, especially for high-complexity models that require extensive training time.

References

[1] Amani Alahmadi et al. "Passive detection of image forgery using DCT and local binary pattern". In: *Signal, Image and Video Processing* 11.1 (2017), pp. 81–88.

[2] Mohammed Hazim Alkawaz et al. "Detection of copy-move image forgery based on discrete cosine transform". In: *Neural Computing and Applications* 30.1 (2018), pp. 183–192.

[3] Jing Dong and Wei Wang. *CASIA tampered image detection evaluation database*. 2011.

[4] A Jessica Fridrich, B David Soukal, and A Jan Lukáš. "Detection of copy-move forgery in digital images". In: *in Proceedings of Digital Forensic Research Workshop*. Citeseer. 2003.

[5] Zhongwei He et al. "Digital image splicing detection based on Markov features in DCT and DWT domain". In: *Pattern recognition* 45.12 (2012), pp. 4292–4299.

[6] Choudhary Shyam Prakash et al. "An integrated method of copy-move and splicing for image forgery detection". In: *Multimedia Tools and Applications* 77.20 (2018), pp. 26939–26963.

[7] Yun Q Shi, Chunhua Chen, and Wen Chen. "A natural image model approach to splicing detection". In: *Proceedings of the 9th workshop on Multimedia & security*. 2007, pp. 51–62.