Received November 22, 2019, accepted December 17, 2019, date of publication January 6, 2020, date of current version January 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2964516

Passive Image Forgery Detection Based on the Demosaicing Algorithm and JPEG Compression

ESTEBAN ALEJANDRO ARMAS VEGA, EDGAR GONZÁLEZ FERNÁNDEZ, ANA LUCILA SANDOVAL OROZCO, AND LUIS JAVIER GARCÍA VILLALBA

Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), Faculty of Computer Science and Engineering, Office 431, Universidad Complutense de Madrid (UCM), 28040 Madrid, Spain

Corresponding author: Luis Javier GarcÍa Villalba (javiergv@fdi.ucm.es)

This was supported in part by the European Union's Horizon 2020 Research and Innovation Programme under Grant 700326.

ABSTRACT Multimedia files play an important role in everyday life. Today, the majority of the population owns state-of-the-art cameras integrated into their mobile devices. Technological development not only facilitates the generation of multimedia content, but also the intentional manipulation of it, and this is where forensic techniques of detecting manipulation on images and videos take on great importance. Although historically there has been confidence in the integrity of images, the advance of technology has begun to erode this confidence. This work proposes a digital image authentication method based on the quadratic mean error of the Color Filter Array interpolation pattern estimated from the analysed image. For the evaluation of the proposed method, experiments were carried out with public databases of forged images that are widely developed for research purposes. The results of the experiments demonstrate the efficiency of the proposed method.

INDEX TERMS Blind technique, chrominance, copy-move, digital image, forensics analysis, forgery detection, splicing.

I. INTRODUCTION

For centuries, human beings have always used images to capture the reality that surrounded them, or to modify it, depending on the message they wanted to convey. Although this evolution undoubtedly has a before and after with the creation of photography in the nineteenth century.

> "The excitement that accompanied the invention of photography was the feeling that man for the first time could see the world as it really was" [1].

eThis statement Collier makes about photography may not fit the letter in today's digital age. There are currently a significant number of computer crimes related to the illegal possession, distribution or modification of multimedia content. The alleged use of mobile devices for this purpose makes these devices an important source of evidence, made by which forensic analysis must be able to authenticate the content and examine whether it is original or was manipulated.

However, the manipulation of visual content has not been exclusive to the current digital age. Over time, manipulation has always been present: In painting, the image to be transmitted to the public has been retouched, for example, in 'The Last Judgement', the painter Michelangelo covered the nudity of some figures by order of the Pope. In conventional photography, it was possible to manipulate through splices the negatives of the photographs, for example in Figure 1 the famous photo of Soviet dictator Iósif Stalin along with his commissioner for Internal Affairs, who disappears from the photo by order of Stalin after being executed in 1940.

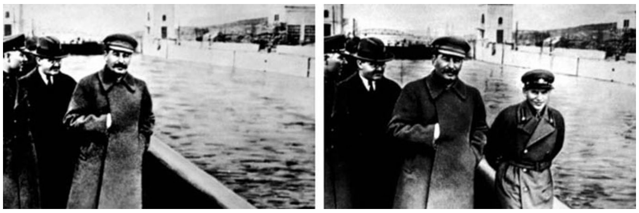


FIGURE 1. Example of manipulation in photography.

While before manipulating visual content had mainly a political, religious or cultural motivation, today, apart from using manipulation for malicious purposes, the most widespread motivation is advertising or aesthetics, for example the Figure 2.

The associate editor coordinating the review of this manuscript and approving it for publication was Tai-Hoon Kim.

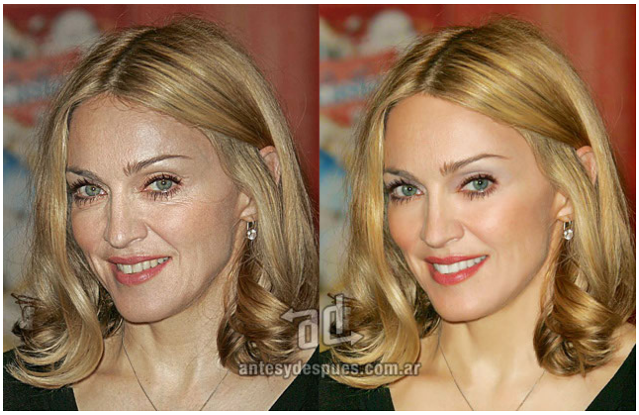


FIGURE 2. Example of manipulation in photography.

The ease of manipulating digital images and videos has increased and is increasing in recent times and is available to the conventional user through programs such as Adobe Photoshop, GIMP, Adobe Premiere, and so on. Even these manipulations are already done automatically by our mobile device through new tools that make use of artificial intelligence such as face enhancers, facial expression changes, improved lighting of the scene, and so on.

In July 2017, researchers from the journal Cognitive-Research [2] used a dataset of 40 scenes, 30 of which were subjected to five different types of manipulation, including physically plausible and implausible manipulations. 707 participants were shown to assess people's ability to detect real-world manipulated scenes. The study found that only 60% of the people were able to detect the fake scenes, and even then, only 45% of them were able to tell exactly where the altered content was. One of the main conclusions from their study was that if people cannot differentiate between real and false details in scenes, manipulations could often modify what we believe and remember.

The growth of the use of digital images and their applications in the modern society combined with the downplayed the expertise require for modifying digital photos by image editing applications have compromised the authenticity and integrity of digital images. Moreover, the simple and fast information exchange through the Internet has caused society to accept much of this material without questioning its integrity.

In this fast digital world, new and advanced forges are conceived every minute, while forensic techniques continue growing to fight against them. However, not all image manipulations are malicious nor dangerous. There are many valid and legal reasons to use sophisticated editing tools to edit or improve images, such as for marketing and design. Unfortunately, all these tools developed for those benign tasks can also be used to manipulate the truth, which can finally alter any legal procedure. It is in these cases when the forensic analysts need to have a set of reliable, updated, and fast tools to define whether or not the authenticity of any image. Despite a large number of tampering techniques, splicing, and copy-move are the most common manipulations. This paper presents two algorithms focused on detecting the presence of both splicing and copy-move, forgeries within an image. Therefore, our primary goal with this work is to help forensic analysts' work by increasing the number of tools and algorithms to improve their results.

This paper presents two algorithms; the first one is an Error Level Analysis (ELA) algorithm, which can be used as an initial filter to detect the presence of splicing in an image. The second algorithm is a digital image authentication method based on the quadratic mean error of the Colour Filter Array (CFA) interpolation pattern estimated from the analysed image. The rest of the work is divided as follows: In Section II, the literature related to manipulation detection techniques are presented. Next, both an Error Level Analysis technique and a chromatic interpolation algorithm estimation technique are proposed to identify the modified area within an image in Section III. Section IV describes the experiments carried out to evaluate the efficiency of the proposed techniques, and the results are also discussed. Finally, the Conclusions section shows the main conclusions and future work.

II. BACKGROUND

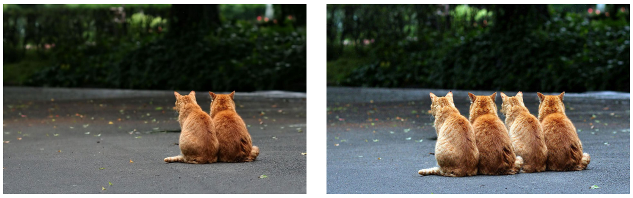
In this section, we will explain the main techniques that are used to manipulate the multimedia content of images. In addition to each technique, a graphic example is shown. At the end of this section, we present a comparative table of the most commonly used image and video editing software tools.

A. COPY-MOVE FORGERY TECHNIQUE

Copy-Move manipulation is typically done to make an object 'disappear' from the original image by covering it with a small fragment copied from another part of the same image. This method is also used to duplicate existing objects in the picture. As these copied blocks come from the same image, all their features will be compatible with the rest of the content, so it is very difficult for the human eye to detect them.

When the copied region is moved, it is usually accompanied by a blurring effect generally used on the edges of the modified region in order to diminish the irregularities between the original and manipulated region. Mainly, the detection techniques for this type of manipulation focus on the search for duplicate areas, although if combined with other post-processing techniques, such as the application of colour filters, or geometrical transformation, it can make detection by existing methods quite tricky. The use of this technique is shown in Figure 3. In the manipulated Figure 3(b) the two animals that appeared in Figure 3(a) have been duplicated.

The main evidence usually used to detect copy-move manipulations is the existence of two equal areas based on the properties of the blocks into which the image is divided. One of the first approximation to identify copied areas, within images, was made in 2003, in [3] Fridrich *et al.* proposed a method that made use of the Discrete Cosine Transform (DCT) coefficient characteristics from overlapping



(a) *Original Image* (b) *Copy-Move Image Forged*

FIGURE 3. Copy-move forgery technique.

blocks of the image. Fridrich *et al.* present one of the first method that use DCT to detect copy-move forgeries on images.

In [4], Popescu *et al.* proposed an algorithm to identify duplicate regions in a digital image. Their algorithm applies principal components analysis (PCA) instead of DCT. The algorithm applies PCA on small fixed-size image blocks, and each block is then lexicographically sorted. This proposed technique has shown high efficiency to detect copy-move forgeries and also, show that the detection is possible even in the presence of significant amounts of corrupting noise.

In [5], the use of the Singular Value Decomposition (SVD) was proposed to distinguish the altered areas in a digital image in 2008. With SVD, the feature vector is also extracted, and the dimensions thereof reduced. Similar blocks were identified through the use of lexicographic classification. This method proved to be robust and efficient. The experimental results demonstrate the validity of the proposed approach for manipulated images subjected to Gaussian blur filters, noise contamination and compressions.

In [6] proposed to detect forgery copy-move in digital images using the Scale-invariant Feature Transform (SIFT) algorithm in 2009. The authors presented the SIFT calculation algorithm using the block matching function. This algorithm offers excellent results even when the image is compressed or noisy.

In [7] a scheme based on Speeded Up Robust Features (SURF) was proposed, which have key point characteristics better than SIFT because they work better with post-processing techniques such as brightness and blur variations. However, the methods based on key points present a problem of visual output because the copied and pasted regions consist of lines and points that do not show a clear and intuitive visual effect.

In [8], Amerini *et al.*, introduced a method based on the scale-invariant feature transform (SIFT). This method can detect duplicated regions in images. Also, the method proposed can detect which geometric transformation was applied. Due to the copied region of the image looks the same as the original, the key points extracted in the duplicated region will be identical to those in the original. This method is also useful with low-quality factor compressed images.

Muhammad *et al.* in [9] presented a blind copy-move forgery detection method based on the dyadic wavelet transform (DyWT). The algorithm uses mainly two kinds of information, the similarity between blocks of the image and the noise inconsistency between these parts. The experiments were executed in three different scenarios: i)same size images, and copy-move region without rotation, ii)different size images and copy-move region with and without rotation, iii)different quality (Q) factors. The results have shown that the algorithm works better than some previous proposals.

In [10], Zhao *et al.*, proposed a method based on discrete cosine transform (DCT) and singular value decomposition (SVD) which include seven steps to analyze and detect duplicate regions in images. In the beginning, the input image is divided into overlapping blocks, then DCT is applied to each block, and the DCT coefficients are quantized. Later, each quantized block is divided non-overlapping sub-blocks, and SVD is applied to each sub-block, then features are extracted to reduce the dimension of each block using its largest singular value. In the end, all feature vectors are lexicographically sorted, and duplicated image blocks will be matched by the predefined threshold. The experiment has shown that the proposed algorithm not only detect copy-move forgery and locate the duplicated regions, but also can analyze and detect manipulation over images with Gaussian blurring, additive white Gaussian noise, and JPEG compression.

In [11], Park *et al.* proposes an approach that can manage several geometric transformations, including rotation, scaling and reflection. The proposed algorithm use keypoints and descriptors from the image based on the Scale Invariant Feature Transform (SIFT) to analyze possible reliable matched pairs by using the distance ratio between the most and second similar match. The matched pairs are then included in a set of real-matches sorted by their ratio value.

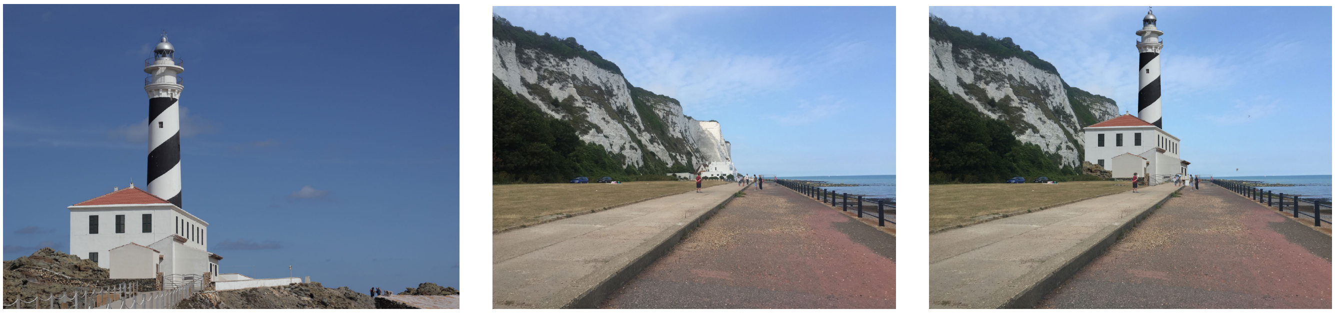
B. SPLICING FORGERY TECHNIQUE

Image splicing is one of the simplest and most commonly used manipulation schemes. This manipulation is similar to the Copy-Move technique, but with the difference that the fragment that is copied does not belong to the same image, but to a different one, that is, the manipulated image is the result of the mixture of two or more images. The objective of this technique is to insert elements that were not in the scene that was originally captured.

As a general rule, the "*donor*" image block may have been acquired by another mobile device and therefore its characteristics and traces will be different from the rest of the image. Detection of this type of manipulation is a fundamental task during image integrity verification. An example of this technique is shown in Figure 4. The Figure 4(a) is the "*donor*" image, the lighthouse is copied and pasted into the "*recipient*" image Figure 4(b). As a result, the splice, Figure 4(c) is generated.

In general, all the detection techniques are based on the variations found in the pattern of characteristics of the pasted area with respect to the content of the "*recipient*" image.

In [12], Shi *et al.*, proposed a blind, passive image splice detector method. This method, extract statistical features from the images, and also, the resulting features of applying a multi-size block discrete cosine transform (MBDCT). These



(a) *Image Original "donor"* (b) *Image Original "recipient"* (c) *Image Result*

FIGURE 4. Splicing technique.

two groups of features build the feature vector that will be the input for the SVM classifier. The experiments carried out by the authors show a higher detection rate, up to 90% accuracy. The public dataset used during their experiments was "*Columbia* [13]".

In [14] Zhang *et al.*, present an algorithm to classify spliced images. The author's algorithm uses the characteristics extracted from 2D matrices generated when applying MBDCT [12]. Their work, beside previous researches, introduce as features, the image quality metrics (IQM). The new vector build from all those features is the input for the SVM classifier. The dataset used for the experiments was "*Columbia*". The obtained accuracy was up to 87.10%

Wang *et al.*, in [15] proposed a passive image tampering detection method based on modeling edge information. Because the human eye is more sensitive to the luminance component (Y) than the chroma component, some tampering artifacts left in the chroma channel are undetectable at first sight. Therefore, the Wang *et al.*'s algorithm transforms the image from RGB to YCbCr space colour and uses only the Cb and Cr components to extract the edge information. Moreover, a finite-state Markov chain (MC) is used to model the thresholded edge image and to capture its interpixel dependencies. Once that the features are extracted, a nine dimensions vector is build to be the input of the SVM classifier. The experiments carried out have shown that the proposed algorithm is very useful for tampering detection. The accuracy obtained was up to 95.6% with the public dataset "*CASIA TIDE v2.0* [16]".

In [17], Zhao *et al.* compared the effect of using different space colors on the detection of image splicing. The authors made a comparison of the YCrCb space color against the regularly used RGB. The algorithm extract and use four gray level run-length run-number (RLRN) vectors from the chroma channels. After the feature extraction, the resulting vector is the input to an SVM, which is the algorithm classifier. The experiments used the datasets "*CASIA TIDE v1.0*" and "*Columbia*" and the detection effectiveness was up to 94.7% of accuracy. The results show that the chrominance channels are more effective than RGB in detecting forgery within images.

Xia *et al.*, in [18], introduced an algorithm to identify forgery within fingerprints images. To extract the needed features to build the input vector for the classifier, Xia *et al.*'s algorithm uses the discrete wavelet transform (DWT) and local binary pattern (LBP). The accuracy obtained by the experiments shown up to 92%. The images used in [18] are in the "*LivDet*" [19] dataset.

In [20] Alahmadi *et al.*, presented a method based on discrete cosine transform (DCT) and local binary pattern (LBP) to detect splicing and copy-move counterfeits. Their algorithm pre-processes the image by changing the space color to YCbCr. Then, divide the Cb and Cr components into overlapping blocks and to each block apply LBP. Each block is transformed into the DCT domain and extract their DCT coefficients to build the feature vector. As a classifier, the authors used an SVM. The experiment results show an accuracy of detection up to 97.77%. The used dataset was "*CASIA TIDE v2.0*".

III. DESCRIPTION OF THE PROPOSED TECHNIQUES

This section will describe the proposed techniques for detecting counterfeits in colour images.

A. ERROR LEVEL ANALYSIS DETECTION TECHNIQUE

The ELA technique focuses on the identification of areas with different levels of compression within the same image. A compressed image in JPEG format must have approximately the same level in all its content. If there is an area with a significantly different error level, then it has a high probability that there has been a digital manipulation of it.

It could be said that ELA highlights the areas of the image most likely to degrade their colours in the next re-compression because the edited areas have a greater potential for degradation compared to the rest of the image.

The algorithm JPEG operates on an 8×8 pixel array, and each 8×8 square is compressed independently. If the image is not completely modified, all 8×8 squares should have similar error potentials, in other words, that when re-compressed each square will degrade at approximately the same speed. ELA re-compresses the image at a specific quality level. This re-saved therefore introduces a known amount of error throughout the image that is compared to the original image. If the image is modified, each affected 8 × 8 square should have a higher error potential than the rest of the image, so the modified areas will appear with a higher level of potential error.

Our proposed algorithm aims to detect areas of the image that do not belong to the original content. It is developed in Python 2.7, using libraries specialised in image processing such as OpenCV and PIL. The input and output of the program is an image.

According to the ELA technique, explained above, an original image JPEG should have the same level of compression throughout its entire content. When the image contains a region that does not belong to its original content, the borders and textures of that area will be highlighted from the rest. Also, taking into account that the compression JPEG is adjustable we can know the compression level of the image content

The algorithm needs two inputs. The first entry in the program is a directory containing one or more JPEG images. For optimal results, it would be desirable that these images have the highest possible resolution and an integer between 0 and 100 that represents the level of JPEG compression we want to use. The most recommended is between 85 and 95.

For each image, a thread is launched that will treat the image and generate its output. The first step to analyse the image is to re-compress it in JPEG format with the quality level declared as an input parameter. Once you have the input image and its re-compression, you must obtain the difference pixel by pixel and in absolute value between both images. With this result, it is possible to identify which pixel area has undergone a more significant change when applying the compression level recovering the maximum and minimum values obtained. The values of the pixels that make up the program's output image are calculated based on the maximum and minimum values previously calculated. To do this, they are scaled based on the 255.0 RGB value, and the brightness of each is enhanced.

Finally, a mask is applied to the generated image to highlight all the areas that have been left with blue and red tones with more brightness than the rest of the content. As the mask covers the less bright areas, the RGB image is converted to the HSV color model. The goal of this conversion is because working with HSV values makes it easier to isolate colors. In the color representation HSV, the hue determines the color you want, the saturation determines how intense the color is, and the value determines the clarity of the image. To isolate colors, multiple masks must be applied. A low threshold mask and a high threshold mask for hue, saturation and value. Any pixels within these thresholds will be set to 1 and the remaining pixels will be zero. These thresholds are configurable at the code level. The conversion from RGB to HSV is governed by the following formulas:

$$R' = \frac{R}{255.0}, \quad G' = \frac{G}{255.0}, \; B' = \frac{B}{255.0} \tag{1}$$

$$C_{max} = max(R', G', B'), \quad C_{min} = min(R', G', B') \tag{2}$$

$$\delta = C_{max} - C_{min} \tag{3}$$

$$H = \begin{cases} 60 \circ \dfrac{G' - B'}{\delta} mod\ 6 & C_{max} = R' \\ 60 \circ \dfrac{B' - R'}{\delta} + 2 & C_{max} = G' \\ 60 \circ \frac{R' - G'}{\delta} + 4 & C_{max} = B' \end{cases} \tag{4}$$

$$S = \begin{cases} 0 & C_{max} = 0 \\ \dfrac{\delta}{C_{max}} & C_{max} \neq 0 \end{cases}$$

$$V = C_{max} \tag{5}$$

Once the mask has been applied, the output image of the program is left with the areas affected by splice marked with a white colour that stands out from the rest of the content because it is either a black area or areas with white pixels, but is isolated.

The image is saved in the output directory so that the researcher can check those areas that stand out most and compare them with the input image to verify whether or not that area belongs to the original content.

B. COLOUR FILTER ARRAY DETECTION TECHNIQUE

The first step of this technique is to estimate the interpolation pattern of the colour filter matrix of the digital camera that captured the image. For this process the image is re-interpolated with various CFA patterns. For each pattern we get its Mean Square Error (MSE) between the original image and the re-interpolated image.

The results obtained from the MSE are then analysed to determine whether the image has been modified. It is expected that one of the values of the calculated SSM for each CFA standard will be much smaller than the other three. If none of the four values is significantly smaller than the others, it can be inferred that the image may have been post-processed. However, at this point you cannot be sure what type of modification has been made or whether it has been retouched.

Being $L_c(x, y)$ the intensity of the colour channel image c in a spatial location (x, y) y $c \in \{R, G, B\}$, the next step is to define the colour filter mask that is done as shown in Eq. 6.

$$\theta_{k,c}(x, y) = \begin{cases} 1, & (x, y) \in \psi_{k,c} \\ 0, & \text{other case} \end{cases} \tag{6}$$

where, $\psi_{k,c}$ represents the location of the array of channel colour filters set c for a particular type of CFA pattern denoted by k and $\theta_{k,c}.(x, y)$ the corresponding colour filter mask of $\psi_{k,c}$.

The technique uses blocks of size $W \times W$, where $W = 8$ pixels, to divide the image taking into account only non smooth blocks. Each non-smooth block is denoted as B_i where $i = 1, \ldots, N.$, being N the number of non-smooth blocks contained in the image. Blocks re-interpolated with the k filter are denoted as $\hat{B}_{i,k}$. These blocks are calculated by a convolution between the bilinear kernel and the re-displayed block B_i with the *kth* CFA pattern defined with Eq. 7.

$$\hat{B}_{i,k} = f(B_i, \theta_k) \quad k = 1, \ldots, 4 \tag{7}$$

Next, calculate the MSE error between the blocks of B and $\hat{B}$ in non-smooth regions over the entire image using Eq. 8.

$$E_i(k, c) = \frac{1}{W \times W} \sum_{x=1}^{W} \sum_{y=1}^{W} (B_i(x, y, c) - \hat{B}_{i,k}(x, y, c))^2 \tag{8}$$

where, E_i is an array containing the average quadratic errors for each colour channel.

To detect the relative distances between the colour channels a new error matrix E_i^2 is created. The normalisation of all rows of the E_i array is done with Eq. 9.

$$E_i^{(2)}(k, c) = 100 \times \frac{E_i(k, c)}{\sum_{l=1}^{3} E_i(k, l)}, \quad c = 1, \ldots, 3 \tag{9}$$

Because there are fewer pixels interpolated in the green channel, the values of the green channel column $V_i(k)$ are taken to determine if there is any modification. This process is done with Eq. 10.

$$V_i(k) = 100 \times \frac{E_i^{(2)}(k, 2)}{\sum_{l=1}^{4} E_i^{(2)}(l, 2)} \tag{10}$$

By means of the uniformity of the vector V_i a possible post-processing operation can be indicated. The uniformity of the green channel vector is defined with Eq. 11.

$$U(i) = \sum_{l=1}^{4} |V_i(l) - 25| \tag{11}$$

Finally, the median of the U vector is calculated as a CFA filter tracking metric as shown in Eq. 12.

$$F = median(U) \tag{12}$$

The higher the CFA filter metric (F), the more likely it is that the image can be interpolated with the CFA filter. Therefore, it can be inferred that no significant processing or alteration occurred. Another way to measure the artifacts of the CFA chromatic interpolation algorithm is to observe the changes in the power of the sensor noise in the given image. If an image is interpolated, the sensor noise in the interpolated pixels is expected to be suppressed. This is due to the nature of the low-pass interpolation. The variance of sensor noise in interpolated pixels becomes significantly lower than the noise power of the sensor in non-interpolated pixels. Therefore, the artifacts of the interpolation algorithm can be measured by comparing the ratio of noise variances of interpolated and non-interpolated pixels. If this ratio is close to 1, it can be assumed that the input image was manipulated.

A typical way to obtain sensor noise is through the wavelet-based noise elimination algorithm presented in [21], [22]. This process is done on the green channel of an image by separating the interpolated pixels from the non-interpolated pixels using the green channel filter mask $\theta_{k,c}$, where $k = 1$ and $c = 2$.

TABLE 1. Datasets features.

Datasets	Format	Resolution	Number of Images
CASIA v1.0 [16]	JPEG	384x256	921 (splicing: 451)
Own [16]	JPEG	1080x1920	30

TABLE 2. Characteristics of the experimental equipment.

Resources	Features
Operating System	Ubuntu 18.04
Memory	4 GB
Process	Intel® Core™ 2 Quad CPU Q8200 @ 2.33GHz x 4
Graphics	NV96
HDD	100 GB

The non-interpolated pixels are divided into 2 vectors A_1 and A_2 to obtain the ratio of the variations of the sensor noise to Eq. 13.

$$F_2 = max(\frac{var(A_1)}{var(A_2)}, \frac{var(A_2)}{var(A_1)}) \tag{13}$$

where *var* represents the variance of the vector and *max* returns the highest value between x and y.

IV. EXPERIMENTS AND RESULTS

A. ERROR LEVEL ANALYSIS EXPERIMENTS

For the evaluation of this algorithm, the public dataset CASIA v1.0([16]) has been used. This dataset contains images manipulated by cropping and pasting operations using Adobe Photoshop CS3 version 10.0.1 in Windows XP. The spliced regions are from the same authentic image (copy-paste) or from another image (splice). This is why only those dataset images containing the spliced region from a different image will participate in this experiment, as the algorithm is designed for splice detection. A specific dataset has also been generated for this experiment with high resolution manually spliced images. Table 1 shows a summary of the characteristics of the dataset used in the experiment.

The characteristics of the equipment with which the experiments were carried out are presented in Table 2. This is an important factor to bear in mind since the execution times of the different tests vary according to the computational resources available.

The experiment carried out on this work was based on the verification of images resulting from the application of the splice detection algorithm in digital images. This algorithm has been applied to the splice images of the CASIA v1.0 dataset for later revision. We have also used our own small dataset with images taken by iPhone that have been manually spliced. The review consists of comparing with the naked eye if the regions that stand out the most in the resulting image are the regions that have suffered the splice. Figure 5 shows the positive results and the Figure 6 shows a bad result of the algorithm. Table 3 shows the results obtained.

B. COLOUR FILTER ARRAY EXPERIMENTS

To evaluate the efficiency of the method described, we used the images of the datasets [23] and [24], denominated D1 and D2 respectively.

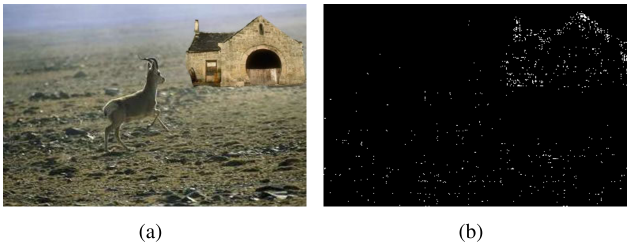


FIGURE 5. Positive example: The white area that stands out most in (b) corresponds to the region pasted in (a). The rest of the white pixels, being isolated, should not be taken into account.

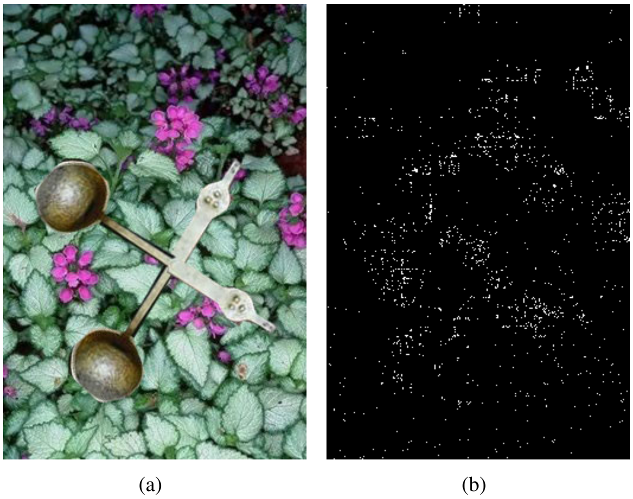


FIGURE 6. Negative example: The white pixel zones in (b) do not clearly distinguish which zone has been pasted in (a).

TABLE 3. Detection of positives after applying the algorithm.

Datasets	Number of Images	Positive	Performance
CASIA v1.0	451	248 / 55%	00:00:25s
Own	30	22 / 73.3%	00:02:10s

D1 dataset images have the following characteristics: High resolution images (3000 × 2000 or 2000 × 3000 pixels minimum), with realistic copy and move forgeries ("realistic" refers to the amount of pixels copied, the treatment of the pixels of the border of the copied region and the content of the region).

D2 dataset images have the following characteristics: The resolution of the images are of medium size (1000 × 700 or 700 × 1000), with uncompressed images with simply copied and moved regions, uncompressed images with simple scenes (an object, simple background) instead of complex scenes, since the dataset is used to study mainly the robustness against some specific attacks.

The characteristics of the equipment with which the experiments were carried out are presented in Table 2. This is an important factor to bear in mind since the execution times of the different tests vary according to the computational resources available.

To evaluate the efficiency of the described method, high resolution images (greater than 1500 × 1500 pixels) were used with and without alterations in different areas of the image [23], [24]. In addition, the time it takes for the method to show the area where it has been modified was measured. In the Figure 7 you can see that the result obtained. Figure 7(a) shows the original image, Figure 7(b) shows the modification made and Figure 7(c) shows the result obtained when applying the proposed technique. It shows the region where the alteration was applied by highlighting the modified area. It should be noted that the dimensions of the image are 2000 × 3008.

However, there are cases where the results are not as clear due to the conditions of the image, for example when there are very clear backgrounds such as skies causing areas to be marked where there is no modification.

The Figure 8 shows this, although it does the delimitation of the modified area correctly, zones are shown in the part of the sky (Figure 8(c)) where calculations indicate that there is a forgery.

When analysing the results with small images from the D2 dataset (see Figure 9) it could be observed that the method is not accurate due to the low resolution of the image and that when processing the image and forming the blocks of size $W \times W$, described in previous sections, the lack of information in the image makes all the variances low and there is no significant difference between them.

Table 4 shows the time it took for the method to analyse images of different resolutions. The time taken to process high resolution images was 24.2959 seconds which shows that the method is efficient and very accurate with large images.

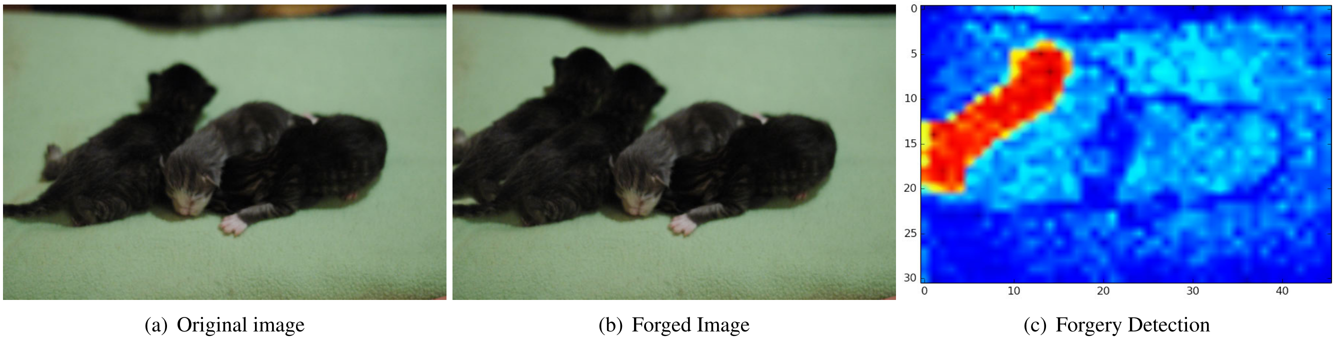
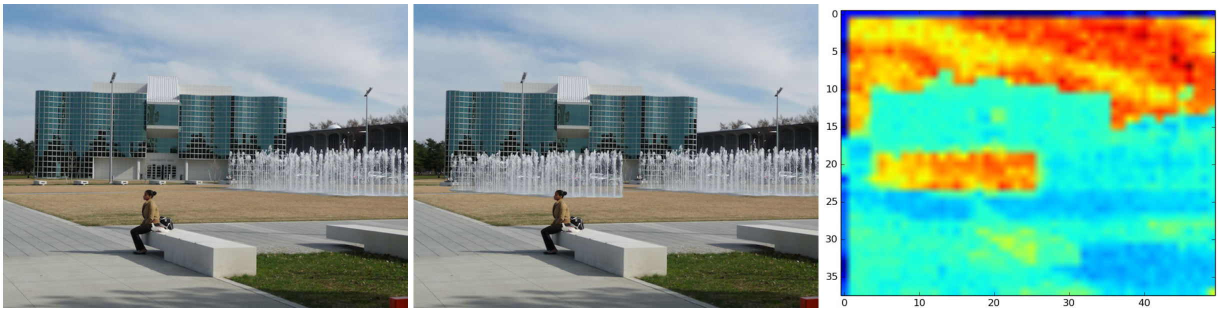
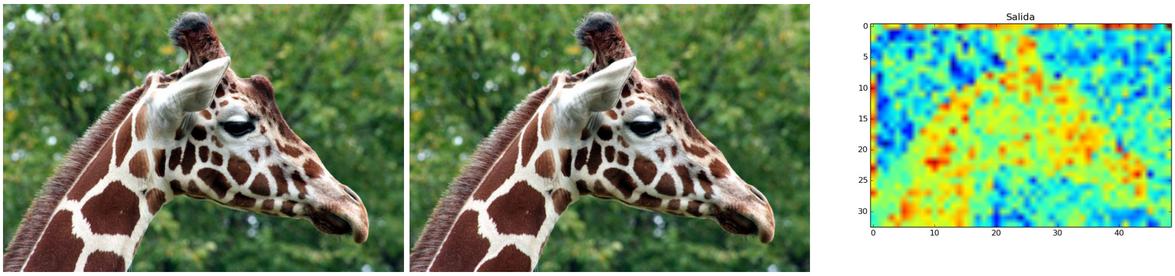


FIGURE 7. Optimal results.



(a) Original Image (b) Forged Image (c) Forgery detected with false positives

FIGURE 8. Results with errors.



(a) Original Image (b) Forged Image (c) Forgery detected with false positives

FIGURE 9. Results over low resolution images.

TABLE 4. Method execution time.

Resolution	2000x3008	2014x3038	2304x3072	2448x3264
Time (s)	20.3427	25.9153	22.6366	28.2889

The proposed method was developed in the Python programming language as it has libraries that facilitate image processing. The processing time of each image is low considering that all the information of the image is used without any previous processing and that the test images are high resolution and color images.

V. CONCLUSION

The content of digital images possesses information that goes beyond the visual. This information is of great forensic value since its correct exploitation can guarantee the authenticity and integrity of the content. Because of this, digital images are an excellent source of evidence when it comes to resolving legal proceedings. The development and continuous improvement of new technologies enable conventional users to be able to alter the content of images and videos with professional results, invisible to the human eye. This is added to the fact that the detection of manipulations is a complex task and also requires continuous improvement to adapt to such a scenario, so it is essential to develop forensic tools capable of detecting these manipulations, increasingly professional and common. The line of research that has been followed in this work begins with a study of the existing techniques of detection of manipulation on images dedicating more effort to techniques of detection of copy-move and splicing in images.

Two forgery detection techniques have been designed and developed: Firstly, an algorithm based on compression JPEG for splice detection in images. This algorithm uses the Error-level-Analysis technique and highlights those pixels with a different compression level. Finally, areas that do not belong to the original content are marked in the image. Secondly, a technique to detect forgery in a colour image using chromatic interpolation algorithms. In the development of the work, it could be observed that by estimating the interpolation pattern and the mean quadratic error of blocks in the image, it can be determined whether or not there is a modification in a given image.

For the first detection technique, the CASIA v1.0 public dataset and an own dataset have been used. The results show that they depend directly on the quality of the image because, for CASIA, the algorithm presents difficulties in detecting the region that does not belong to the original image. However, the dataset itself contains high-resolution images where a 73.3% of accuracy was obtained. It is important to mention that the accuracy of the result is determined by the analyst since it is he who has to consider whether the spliced area is clearly highlighted in the image obtained. Therefore, it is necessary to know well how ELA works. However, this technique serves as the first evidence, since, being a rapid response detection technique, it can serve as a source of suspicion for a researcher to further investigate those images that have presented suspicious white regions.

For the CFA algorithm, satisfactory results were obtained when entering images with dimensions greater than 1500 × 1500 pixels delimiting the modified area. Images that have a big area with white colors create false positives because the variance calculated in these sections is very low to the rest of the image. Likewise, the best results are obtained with large images since the image information is sufficient to calculate the variance of the image correctly, and a distinction can be made between them. However, with images smaller than 700 × 700 pixels, the method has difficulties in detecting the area with modifications since the image information is not sufficient to make the difference between the variances known.

REFERENCES

[1] G. Winskel and M. Nielsen, *Principles of Visual Anthropology*. London, U.K.: Univ. of Cambridge, 1975, pp. 211–230.

[2] S. J. Nightingale, K. A. Wade, and D. G. Watson, "Can people identify original and manipulated photos of real-world scenes?" *Cogn. Res. Princ. Implications*, vol. 2, no. 1, p. 30, Jul. 2017.

[3] J. Fridrich, D. Soukal, and J. Lukas, "Detection of copy move forgery in digital images," in *Proc. Digit. Forensic Res. Workshop*, Binghamton, NY, USA, Aug. 2003, pp. 5–8.

[4] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Dept. Comput. Sci., Dartmouth College, Hanover, NH, USA, Tech. Rep. TR2004-515, Jan. 2004, vol. 646.

[5] X. Kang and S. Wei, "Identifying tampered regions using singular value decomposition in digital image forensics," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, vol. 3, Dec. 2008, pp. 926–930.

[6] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using SIFT algorithm," in *Proc. IEEE Pacific–Asia Workshop Comput. Intell. Ind. Appl.*, vol. 2, Dec. 2008, pp. 272–276.

[7] X. Bo, W. Junwen, L. Guangjie, and D. Yuewei, "Image copy-move forgery detection based on SURF," in *Proc. Int. Conf. Multimedia Inf. Netw. Secur.*, Nanjing, China, Nov. 2010, pp. 889–892.

[8] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra, "A SIFT-based forensic method for copy-move attack detection and transformation recovery," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1099–1110, Sep. 2011.

[9] G. Muhammad, M. Hussain, and G. Bebis, "Passive copy move image forgery detection using undecimated dyadic wavelet transform," *Digit. Invest.*, vol. 9, no. 1, pp. 49–57, Jun. 2012.

[10] J. Zhao and J. Guo, "Passive forensics for copy-move image forgery using a method based on DCT and SVD," *Forensic Sci. Int.*, vol. 233, nos. 1–3, pp. 158–166, Dec. 2013.

[11] C.-S. Park and J. Y. Choeh, "Fast and robust copy-move forgery detection based on scale-space representation," *Multimedia Tools Appl.*, vol. 77, no. 13, pp. 16795–16811, Jul. 2018.

[12] Y. Q. Shi, C. Chen, and W. Chen, "A natural image model approach to splicing detection," in *Proc. 9th Workshop Multimedia Secur. (MM&Sec)*, Dallas, TX, USA, Sep. 2007, pp. 51–62

[13] Columbia University. *Columbia DVMM Image Splicing Datasets*. Accessed: Jun. 2019. [Online]. Available: http://www.ee.columbia.edu/ln/dvmm/newDownloads.htm

[14] Z. Zhang, J. Kang, and Y. Ren, "An effective algorithm of image splicing detection," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, vol. 1, Dec. 2008, pp. 1035–1039.

[15] W. Wang, J. Dong, and T. Tan, "Image tampering detection based on stationary distribution of Markov chain," in *Proc. IEEE Int. Conf. Image Process.*, Hong Kong, Sep. 2010, pp. 2101–2104.

[16] J. Dong and W. Wang. *CASIA TIDE V1.0 and V2.0*. Accessed: Jun. 2019. [Online]. Available: http://forensics.idealtest.org/

[17] X. Zhao, J. Li, S. Li, and S. Wang, "Detecting digital image splicing in chroma spaces," in *Digital Watermarking*, vol. 6526. Berlin, Germany: Springer, 2011, pp. 12–22.

[18] Z. Xia, C. Yuan, X. Sun, D. Sun, and R. Lv, "Combining wavelet transform and LBP related features for fingerprint liveness detection," *IAENG Int. J. Comput. Sci.*, vol. 43, no. 3, pp. 290–298, Apr. 2016.

[19] Listverse. (Oct. 2007). *Top 15 Photoshopped Photos That Fooled Us All*. [Online]. Available: http://listverse.com/2007/10/19/top-15-manipulated-photographs/

[20] A. Alahmadi, M. Hussain, H. Aboalsamh, G. Muhammad, G. Bebis, and H. Mathkour, "Passive detection of image forgery using DCT and local binary pattern," *Signal, Image Video Process.*, vol. 11, no. 1, pp. 81–88, Jan. 2017.

[21] A. E. Dirik and N. Memon, "Image tamper detection based on demosaicing artifacts," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 1497–1500.

[22] A. L. S. Orozco, J. Hernandez-Castro, L. J. G. Villalba, S. J. Gibson, D. M. A. González, and J. R. Corripio, "Smartphone image acquisition forensics using sensor fingerprint," *IET Comput. Vis.*, vol. 9, no. 5, pp. 723–731, Oct. 2015.

[23] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou, "An evaluation of popular copy-move forgery detection approaches," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 6, pp. 1841–1854, Dec. 2012.

[24] E. Ardizzone, A. Bruno, and G. Mazzola, "Copy–move forgery detection by matching triangles of keypoints," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 10, pp. 2084–2094, Oct. 2015.

ESTEBAN ALEJANDRO ARMAS VEGA received the Computer Science degree from the Polytechnic Institute José Antonio Echeverría, in Havana, Cuba, in 2009, and the M.Sc. degree in computer science from the Universidad Complutense de Madrid, Spain, in 2016. He is currently the Ph.D. degree with the Department of Software Engineering and Artificial Intelligence, Faculty of Computer Science and Engineering, Universidad Complutense de Madrid (UCM). He is also a member of the Complutense Research Group of Analysis, Security and Systems (GASS). His research interests include computer networks and computer security.

EDGAR GONZÁLEZ FERNÁNDEZ was born in Mexico City. He received the degree in applied mathematics from the Universidad Autónoma del Estado de Hidalgo, in 2010, and the Master in Science degree (with specialization in mathematics) from the Center for Research and Advanced Studies of the National Polytechnic Institute (CINVESTAV-IPN), where he is currently pursuing the Ph.D. degree with the Computer Science Department. He is also a member of the Research Group of Analysis, Security and Systems (GASS), Universidad Complutense de Madrid (UCM). His research interests are cryptography, information security, and data science.

ANA LUCILA SANDOVAL OROZCO was born in Chivolo, Magdalena, Colombia, in 1976. She received the Computer Science Engineering degree from the Universidad Autónoma del Caribe, Colombia, in 2001, the Specialization Course in Computer Networks from the Universidad del Norte, Colombia, in 2006, and the M.Sc. degree in research in computer science and the Ph.D. degree in computer science from the Universidad Complutense de Madrid, Spain, in 2009 and 2014, respectively. She is currently a Postdoctoral Researcher with the Universidad Complutense de Madrid, Spain. Her main research interests are coding theory, information security, and its applications.

LUIS JAVIER GARCÍA VILLALBA received the Telecommunication Engineering degree from the Universidad de Málaga, Spain, in 1993, and the M.Sc. degree in computer networks and the Ph.D. degree in computer science from the Universidad Politécnica de Madrid, Spain, 1996 and 1999, respectively. He was a Visiting Scholar with the Computer Security and Industrial Cryptography (COSIC), Department of Electrical Engineering, Faculty of Engineering, Katholieke Universiteit Leuven, Belgium, in 2000, and a Visiting Scientist with the IBM Research Division, IBM Almaden Research Center, San Jose, CA, USA, in 2001 and 2002. He is currently an Associate Professor with the Department of Software Engineering and Artificial Intelligence, Universidad Complutense de Madrid (UCM), and the Head of Complutense Research Group of Analysis, Security and Systems (GASS), School of Computer Science, UCM Campus. His professional experience includes research projects with Hitachi, IBM, Nokia, and Safelayer Secure Communications.