Identification of Cut & Paste Tampering by Means of Double-JPEG Detection and Image Segmentation

M. Barni, A. Costanzo, L. Sabatini
Department of Information Engineering
University of Siena, Siena, ITALY
Email: barni@dii.unisi.it, andreacos82@gmail.com, larasabatini81@gmail.com

***Abstract*— This paper focuses on images whose content has been modified by means of a cut & paste operation. By relying on an existing scheme for the detection of double JPEG compressed images with desynchronized grids, we propose two algorithms for the detection of image regions that have been transplanted from another image. The proposed methods work whenever the pasted region is extracted from a JPEG compressed image and inserted in a target image that is subsequently compressed with a quality factor larger than that used to compress the source image. The new methods are intended as a complement to previous works relying on the detection of artifacts introduced by double JPEG compression with aligned compression grids. The experiments we carried out show the good performance of the novel schemes, the second one providing better results at a lower complexity thanks to the incorporation within the detection process of some information regarding the actual image content.**

I. Introduction

The development of techniques for the detection of image tampering operations changing the content of an image is getting more and more attention from a forensics point of view. In this framework the detection of cut & paste operations whereby a portion of a source image is copied into a target image plays a crucial role, since this is the most common way of changing the semantic content of an image. In the absence of the source image, the detection of the copied region goes through the identification of statistical anomalies that identify such a region as statistically different from the rest of the image. Possible approaches in this sense include the detection of resampling artifacts [1], anomalies in the interpolation used for color demosaicking [2], statistical differences of image noisiness [3], [4].

In this paper, we present a scheme that relies on the detection of artifacts introduced by double JPEG compression. Specifically, the scheme we propose relies on the availability of an algorithm for the detection of images that have undergone a double JPEG compressed with non-aligned 8×8 grids [5]. The main idea behind the proposed scheme is the following. By assuming that both the original and the target images are available in JPEG format, when a portion of the source image is copied into the target image and then recompressed, the original and copied regions undergo a double JPEG compression. However, it is very likely that the 8×8 grid used for the former compression of the copied region is not aligned with the 8×8 grid used for the latter compression, while this will be the case with the non-tampered parts. By detecting the artifacts introduced by non-aligned double JPEG compression we are then able to identify the tampered region. Note that as opposed to similar works in the scientific literature, we identify tampered regions as those for which a double compression is detected, while most of the existing schemes work the other way round (see [6] for example), since they identify the tampered regions as those that are not judged to be double compressed[1]. However, these two approaches have to be considered complementary rather than competing thus in the present work we will not pause to compare their performances.

A problem with the above approach is that the statistical analysis used to detect the presence of non-aligned double JPEG compression must be applied locally to avoid mixing the statistics of altered and original regions, however it is not clear beforehand how such regions should be formed. We propose two ways to solve this problem.

The first approach is a very simple one: each single 8×8 block is classified by relying on the statistics of a sufficiently large rectangular neighborhood of the block itself. By adapting the size of the neighborhood to the minimum size of tampered regions (that we assumed to be known a priori), we expect that blocks in the inner parts of tampered (resp. non-tampered) regions will be classified correctly. Once a map of the tampered blocks is built, morphological operators are applied to remove isolated blocks and obtain the tampered regions.

The simple approach outlined above has two main drawbacks, first of all it is computationally very expensive since each block must be analyzed separately. Secondly it does not take into account the image content, thus there is no guarantee that the regions classified as tampered correspond to meaningful image objects (as it is likely to be for any practical cut & paste operation). For this reason, the second approach we have tested works by first segmenting the image and then analyzing each segmented region by itself, i.e. we try to apply the statistic analysis to meaningful image regions.

We have conducted several experiments to evaluate the validity of the two schemes we have developed. The results we obtained demonstrate that as long as output of the segmentation algorithm is a good one, the second approach gives better results (especially in terms of reduction of false positive rates) with a much lower complexity.

The rest of this paper is organized as follows. In section II, the algorithm for the detection of double JPEG compression our system relies on is briefly reviewed. In section III, the proposed algorithms are described. The results of the experimental analysis we carried out to measure the performance of the proposed schemes are discussed in section IV. Finally, in section V, we draw our conclusions.

II. Detection of cropped and recompressed images

In this section we describe the algorithm for the detection of double compressed images that lies at the heart of the tamper detection scheme. In general, there are two possible instantiations of this problem. In the simplest case, it is assumed that the first and second JPEG coders work on aligned images, e.g. they apply the block-DCT at the basis of the compression algorithm on the same 8×8 grid. If this is the case, the effect of double JPEG compression is the consecutive quantization of the block-DCT coefficients of the image.

[1]The rationale behind those works is that the processing operations usually associated to cut & paste, like resizing or editing, destroy the traces left by the former compression.

If the quantization steps of the two encoders are different, then some artifacts appear in the histogram of DCT coefficients thus making the identification of double-compressed images possible [7], [8].

The situation is more involved when the two encoders work on non-aligned images, possibly because after the first compression the image has been cropped. In this case the detection of double compressed images goes through the identification of the blocking artifacts introduced by the older compression. Results reported in the literature [5], [9] show that the detection of double compressed images is still possible if the former quantization step is larger than the second one, so that the second compression does not erase the blocking artifacts introduced by the former compression.

Our method for the detection of tampered regions relies on the availability of an algorithm for detecting double compression with de-synchronized grids. Specifically, it relies on the work by Luo et al. [5]. In a nutshell, Luo et al's algorithm for the detection of double compressed images works as follows. The image is partitioned into 8×8 blocks aligned with the image borders. Then for each block the following differences are computed:

$$\begin{aligned} Z'(x,y) &= |A + D - B - C| \\ Z''(x,y) &= |E + H - F - G| \end{aligned} \tag{1}$$

where the notation used in the equation is explained in figure 1 and where (x, y) indicate the coordinates of the pixel A within the 8×8 block (thus $x = 1 \dots 8, y = 1 \dots 8$).

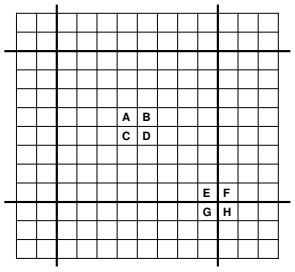

Fig. 1. Position of pixels used for the construction of BACM.

Note that E, F, G and H have a fixed position with respect to A (in particular we have $E = A + (4, 4)$). The values of Z' and Z'' are computed for all the blocks of the image and the corresponding histograms built, for a total of 128 histograms, let us indicate the histograms relative to Z' and Z'' as $H'_{x,y}(n)$ and $H''_{x,y}(n)$ respectively. The absolute difference of H' and H'' is then calculated yielding:

$$K_{x,y}(n) = |H'_{x,y}(n) - H''_{x,y}(n)|. \tag{2}$$

At this point $K_{x,y}(n)$ contains an indication of the difference between the blockiness measured in the group A, B, C, D and that measured in the group E, F, G, H. Such difference is summarized in the BACM (Blocking Artifacts Characteristics Matrix) $M(x, y)$ as follows:

$$M(x,y) = \frac{\sum_n K_{x,y}(n)}{255 \times 2 + 1}. \tag{3}$$

In [5] it is shown that the BACM of images compressed once is very regular with a marked symmetry around the point $(4, 4)$, while the BACM of non-compressed images does not show any particular regularity. In the presence of double compression, we have an intermediate case with the BACM showing some regularity and symmetry, but to a lesser extent than single compressed images. In order to use this property of the BACM for detecting double compressed images, 14 features are extracted highlighting the symmetry of the BACM (see [5] for more details on this point). Such features are fed to a neural network (NN) whose aim is that of distinguishing between single compressed and double compressed images. The final accuracy of the method depends on several factors including the quality factors used for the first and second JPEG compression, the content of the images and their size. Expectedly, better results are obtained when the quality factor used for the former compression is significantly lower than that used for the latter compression. In this case an overall accuracy around 90% can be obtained.

In the sequel, we show how the global algorithm described in this section can be used to build a system capable of detecting cut & paste tampered images.

III. Proposed algorithms for cut & paste detection

In the following we assume that the tampered image is obtained by taking a region R from a source image S and pasting it into a target image T producing a fake image F. We assume that both S and T are available in JPEG format. Moreover we assume that after the insertion of R within T, the tampered image F is again JPEG compressed and stored. We assume that no other processing occurs. It is evident that all the regions in F undergo a double JPEG compression, however in the parts of T that have not been replaced by R, the two subsequent compressions use aligned 8×8 grids. This is not necessarily the case with the region R, since it is unlikely that the cut and paste operation is carried out by paying attention to place the region R in such a way that the old compression grid is aligned with the new image (actually the probability of such an event is 1/64). The presence of a non-synchronized double JPEG compression can then be taken as an evidence of tampering.

As we already observed in the introduction, this is not the only possible approach, in [6], for instance, the absence of synchronized double compression is taken as evidence of region local tampering. We believe that these approaches (and many other available techniques) should be used in a complementary way, since they are based on different complementary assumptions.

The main problem with the above idea is that the possibly tampered regions are not known a priori, hence it is not clear where the statistical information upon which the BACM is built should be gathered. A possibility is to ask the user to highlight the region wherein the statistical analysis should be carried out. While this is surely an admissible solution in some applications, here we aim at devising a completely automatic solution suitable for large scale investigations. For this reason we developed two alternative solutions as detailed in the next subsections.

A. Block-wise approach

The rationale behind the first solution is that each 8×8 block is processed independently by analyzing the statistics of a rectangular region surrounding it. On one side the analyzed region should be large enough to allow a meaningful statistical analysis, while on the other side the region should be small enough so that its characteristics are representative of that of the to-be-classified block. After a thorough experimental analysis we decided to consider a 128×128 region centered on the analyzed block. In practice, for each block a BACM is built by considering the surrounding 128×128 area and the 14 features describing the symmetry of the BACM extracted. Such features are fed to a neural network (NN) trained with a dataset of 5000 JPEG images (2500 compressed once and 2500 compressed twice) generated from a set of 320 not compressed images by means

of random cropping. Compression quality factors QF_1 and QF_2 have been randomly chosen in the intervals $(50, 89)$ and $(60, 95)$ respectively. Note that we did not train the NN to work with a specific second quality factor. However, since this parameter is known, it is reasonable to say that we could improve performances by training several NN's and use the most appropriated one case by case. Specifically we used a three layer NN, with 14 neurons in the first layer, 4 neurons in the hidden layer and one neuron in the output layer. The value of the last neuron is considered as an indication of the probability that the current block belongs to a tampered region. The lower the output of the NN, the more probably the block has been tampered with.

By applying the above procedure to all the blocks of the image, we obtain a tampering map with dark areas corresponding to tampered regions. In order to actually identify such regions, the map is thresholded and the obtained regions processed by means of morphological operators, whose goal is that of removing isolated and small regions and smoothing the shape of the areas identified as tampered.

An example of the results produced by the block-wise detector is reported in figure 2. The original and the tampered images are given on the upper row, respectively on the left and on the right. In the left part of the bottom row the tampering map is given, while the final output with the region identified as tampered is given in the bottom-right part of the figure. In addition to the true tampered areas, a small region below the upper flower is falsely detected. This kind of false alarms could be easily removed either by increasing the minimum size of tampered areas or by visual inspection.

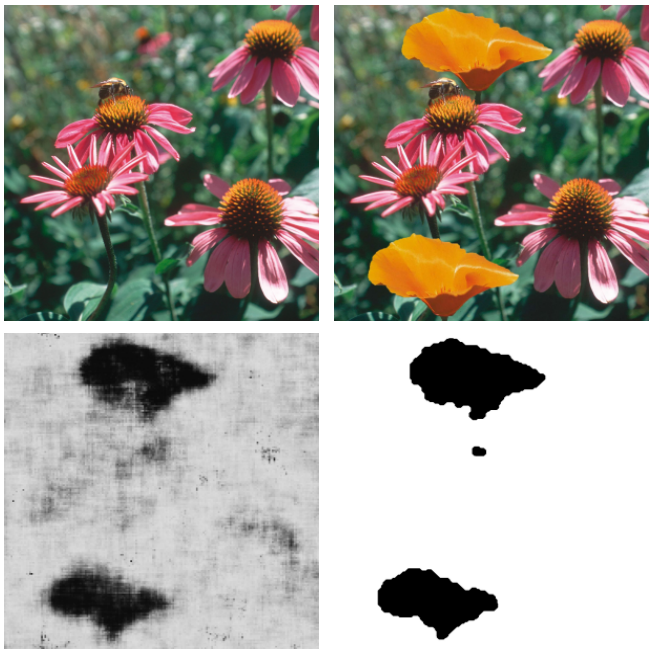

Fig. 2. Example of blockwise tamper detection. Top-left: original image; top-right: tampered image; bottom left: tampering map; bottom-right: tampered region identified by the blockwise detector.

B. Region-wise approach

The block-wise approach has two main drawbacks: first of all it is computationally expensive since each block is analyzed independently. For instance, the non-optimized Matlab implementation we have used in our experiments requires about 22 minutes to process a 1600×1500 image like the one reported in figure 2. The second drawback is that there is no guarantee that the regions identified by the algorithm correspond to meaningful image objects. Yet this would be an important clue to discard false positive errors in case the regions judged as tampered with do not have any meaning, and confirm the decision of the block-wise detector in case such a meaning exists. In order to overcome these problems, we have devised a region-wise detector. The basic idea is again very simple: we first segment the image into homogeneous regions (whatever the term homogeneous may mean) and then analyze each region separately. That is, we build the BACM of each region by analyzing only the blocks belonging to it and use the features extracted from the BACM to classify the whole region at once. In this way, not only computing time is dramatically reduced, but we force the detector to work on meaningful regions. Of course the effectiveness of the region-wise method depends strongly on the performance of the image segmentation algorithm. In our research we focused on completely automatic segmentation, that is no clue about the number of regions or about their position is given to the segmentation algorithm. It goes without saying that better results are likely to be obtained by relying on semi-automatic segmentation in which, for instance, the user provides the seeds of the interesting regions and their number. In our implementation we used a segmentation algorithm derived from the method described in [10] and further refined in [11] [2]. In figure 3, we report the results of the segmentation algorithm and the output of the region-wise detector for the tampered image shown in figure 2. As opposed to the block-wise case, no false alarm is obtained. For the sake of clarity, note that not all the cases of tampering we considered are as simple as the image shown in figure. From a computational point of view, processing such image by means of the region-wise algorithm requires 128 seconds with a dramatic improvement with respect to the block-wise approach.

Fig. 3. Example of regionwise tamper detection. Left: output of the segmentation algorithm; right: output of the tamper detector.

IV. Experimental results

The block-wise and the region-wise algorithms have been tested experimentally on a set of 20 images including both *simple* images with few objects and *complex* images containing many objects and regions. The size of the images ranged from 1024×1024 to 2600×2000 [3]. The original images were JPEG compressed with quality factor equal to 60%. After tampering, the counterfeited images were compressed again with quality factor equal to 90%. For the construction of the

[2]Specifically we used the software freely available for download at http://www.caip.rutgers.edu/riul/research/code/EDISON/index.html.

[3]The dataset used for the experiments is freely available to readers at the address `http://clem.dii.unisi.it/~vipp/files/ISCAS2010`.

tampered images, we automatically pasted non-homogeneous regions in random positions that ensure de-synchronization of the JPEG grids.

TABLE I
PERFORMANCE OF THE BLOCK-WISE ALGORITHM.

False positive	59
False negative	4 (10%)

The results we have obtained by using the block-wise approach are summarized in table I. Specifically the table reports the overall number of false alarms generated by the algorithm and the number of tampered regions that have not been identified by the algorithm. Measuring the number of false alarms is a straightforward task: the block-wise algorithm was applied to the original 20 images of the test set and the number of regions judged as tampered was measured. As to false negative errors, we built 40 tampered images by starting from the 20 images of the test set and run the block-wise algorithm again, then we compared the regions extracted by the algorithm with the actually tampered areas, if less than 50% of the tampered area was detected we considered this result as a false negative error. The number of false positives is rather high, however most of the regions falsely identified as tampered are either rather small or do not correspond to meaningful objects, hence most of these false alarms could be removed by visual inspection.

We then tested the region-wise approach. During the experiments the parameters of the segmentation algorithm were set as specified in table II, however we did not observe a strong dependence of the results upon these parameters, with the only exception of M i.e. the minimum size of the region produced by the segmentation. In fact, M should be tuned to the minimum size of the tampered region, that in principle is not known a priori. In the experiments we let $M = 10000$, a value that permits to collect statistically meaningful features from the segmented regions, and ensures an accurate identification of the main objects contained in the scene.

TABLE II
PARAMETER SETTING FOR THE SEGMENTATION ALGORITHM USED BY THE REGION-WISE DETECTOR. FOR THE MEANING OF THE VARIOUS PARAMETERS WE REFER TO [10], [11]

Basic parameters of mean-shift algorithm [10]	Parameters for the improved version of mean-shift [11]
$h_s = 16$	speed-up = 2
$h_r = 12$	$n = 5$
$M = 10000$	$m = 0.3$
	$t_e = 0.9$

The region-wise approach was tested on the same dataset used for the block-wise algorithm, obtaining the results reported in table III. In this case, the table gives also the number of correct decisions (number of true positive and true negative), since the segmentation of the image permits to count the number of regions processed by the tamper detection algorithm. False negative errors were evaluated as in the block-wise case (but on 60 tampered images instead of 40), i.e. a tampered region was considered to be successfully detected only if at least 50% of its area is identified by the tamper detection algorithm.

By comparing the results produced by the block-wise and the region-wise algorithms we see that the region-wise algorithm allows to reduce both the number of false positive errors (passing from 59 to 40) and the number of false negative events (passing from 10% to 8%). This is an expected effect due to the incorporation within the tamper detection process of information about the actual image content. Finally, we mention the huge advantage of the region-wise approach in terms of computing time. The average processing time for the block-wise approach, in fact, was 18 minutes against an average computing time of 162 seconds for the region-wise algorithm.

TABLE III
PERFORMANCE OF THE REGION-WISE ALGORITHM.

Original images	False positive	40 (8%)
	True negative	460 (92%)
Tampered images	False negative	5 (8%)
	True positive	55 (92%)

V. CONCLUSION

We have proposed two algorithms for the detection of tampered images obtained by means of cut & paste operations. The proposed methods rely on the presence in the tampered areas of artifacts typical of double compressed JPEG images with de-synchronized DCT grids. For these artifacts to be detectable with sufficient accuracy it is necessary that the former JPEG compression uses a lower quality factor than the latter compression. The region-wise algorithm can be seen as a first attempt to couple the tamper detection analysis to the understanding of the semantic content of the image. This is an interesting direction for future research, given that understanding the image content may greatly help to reduce the false alarm rate and/or reduce the complexity of the detector by focusing on a small subpart of the whole image.

ACKNOWLEDGMENT

This work has been partially supported by the project Living-Knowledge - Facts, Opinions and Bias in Time funded by the European Commission under contract no. 231126.

REFERENCES

[1] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of re-sampling," *IEEE Trans. Signal Proc.*, vol. 53, no. 2, pp. 758–767, 2005.

[2] ——, "Exposing digital forgeries in color filter array interpolated images," *IEEE Trans. Signal Proc.*, vol. 53, no. 10, pp. 3948–3959, 2005.

[3] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection with binary similarity measures," in *Proc. European Signal Processing Conf.*, Turkey, 2005.

[4] H. Farid and S. Lyu, "Higher-order wavelet statistics and their application to digital forensics," in *Proc. IEEE Workshop on Statistical Analysis in Computer Vision*, Madison, WI, USA, 2003.

[5] W. Luo, Z. Qu, J. Huang, and G. Qiu, "A novel method for detecting cropped and recompressed image blocks," in *Proc. IEEE Int. Conf. on Acoustic Speech and Signal Processing, ICASSP'07*, vol. II, Honolulu, Haway, USA, April 2007, pp. 217–220.

[6] Z. Lin, J. He, X. Tang, and C.-K. Tang, "Fast, automatic and fine grained tampered jpeg image detection via dct coefficient analysis," *Pattern Recognition*, vol. 42, pp. 2492–2501, 2009.

[7] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *6-th Int. Work. on Inf. Hiding*, Toronto, Canada, 2004, pp. 128–147.

[8] J. Lucas and J. Fridrich, "Estimation of primary quantization matrix in double compressed jpeg images," in *Proc. Digital Forensic Research Workshop*, Cleveland, OH, August 2003.

[9] S. Ye, Q. Sun, and E. C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *Proc. IEEE Int. Conf. Multimedia and Expo*, Beijing, China, 2007, pp. 12–15.

[10] D. Comanicu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, pp. 603–619, May 2002.

[11] C. Christoudias, B. Georgescu, and P. Meer, "Synergism in low-level vision," in *16th International Conference on Pattern Recognition*, vol. IV, Quebec City, Canada, August 2002, pp. 150–155.