A gradient-based pixel-domain attack against SVM detection of global image manipulations

Zhipeng Chen[*,#], Benedetta Tondi[†], Xiaolong Li[*], Rongrong Ni[*], Yao Zhao[*], Mauro Barni[†]

[†]*Department of Information Engineering and Mathematics, University of Siena, Siena, ITALY*
[*]*Institute of Information Science, Beijing Key Laboratory of Advanced Information Science and Network Technology*
Beijing Jiaotong University, Beijing, China
[#] *Department of Computer Science, Tangshan Normal University, Tangshan, China*

chzhpeng@hotmail.com, benedettatondi@gmail.com, lixl@bjtu.edu.cn
rrni@bjtu.edu.cn, yzhao@bjtu.edu.cn, barni@dii.unisi.it,

***Abstract*—We present a gradient-based attack against SVM-based forensic techniques relying on high-dimensional SPAM features. As opposed to prior work, the attack works directly in the pixel domain even if the relationship between pixel values and SPAM features can not be inverted. The proposed method relies on the estimation of the gradient of the SVM output with respect to pixel values, however it departs from gradient descent methodology due to the necessity of preserving the integer nature of pixels and to reduce the effect of the attack on image quality. A fast algorithm to estimate the gradient is also introduced to reduce the complexity of the attack. We tested the proposed attack against SVM detection of histogram stretching, adaptive histogram equalization and median filtering. In all cases the attack succeeded in inducing a decision error with a very limited distortion, the PSNR between the original and the attacked images ranging from 50 to 70 dBs. The attack is also effective in the case of attacks with Limited Knowledge (LK) when the SVM used by the attacker is trained on a different dataset with respect to that used by the analyst.**

I. Introduction

The necessity of understanding the limits of image forensic analysis in the presence of an adversary has prompted the development of a large number of counter-forensic methods [1]. From a general perspective, the recent trend toward the study of the Adversarial Signal Processing from a more theoretical basis goes in this direction [2], [3]. While the first attempts were rather simplistic and boiled down to basic processing like noise addition, recompression, resampling or filtering [4], [5], [6], [7], [8], [9], more recent works aimed at inducing a decision error with the minimum possible distortion so to preserve the quality of the attacked image. This is possible when the attacker has enough information about the details of the forensic algorithm. By adopting the terminology introduced in [10], in a Perfect Knowledge (PK) scenario, the attacker has complete information about the forensic algorithm and hence can optimise the attack in order to minimise the amount of distortion necessary to induce a decision error. In some cases, the attacker only knows the type of detector used by the analyst, e.g. a Support Vector Machine (SVM) or a neural network with known parameters, and the feature space wherein the analysis is carried out, however it does not have access to the data used to train the detector. In this case, referred to as attack with Limited Knowledge (LK) [10], the attacker can build a surrogate version of the detector by using its own training data, and carry out the attack on the surrogate detector, assuming that the attack will also work on the detector used by the analyst.

In principle, the optimum attack could be implemented by applying a gradient-descent technique directly in the pixel domain, however several problems prevent a straightforward adoption of such a strategy. First and foremost, the relationship between the image pixels and the output of the detector, passing through the feature domain, is often very complicated, thus making impossible to derive a closed form expression for the gradient. This is especially true for detectors based on a large number of possibly correlated features like it is customary in forensic detectors based on machine learning [11], [12]. Together with the necessity of computing the gradient for all the pixels of the image, this results in an unmanageable computational complexity.

A possibility to overcome the above problem is to derive the optimum attack in the feature domain and then modify the image pixels so that the features extracted from the attacked image correspond to those defining the optimum attack. In [13], for instance, the optimum attack is derived in the block-DCT domain and the attacked image is obtained by applying the inverse DCT. Such an approach, however, works only when the relationship between the pixel and the feature domain is invertible (as in [13]) and the distortion introduced in the feature domain corresponds to that introduced in the pixel domain or, more in general, when the distortion in the pixel domain can be controlled by operating in the feature domain (once, again this is the case with pixel and DCT domains). Other examples of the above approach are given in [14], [15], [16] where the distortion introduced by attacking the image in the histogram domain is directly related to the distortion introduced in the pixel domain by resorting to optimal transport theory [17], [18]. The tradeoff between effectiveness of the attack and distortion introduced for the case of double JPEG anti-forensics is investigated in [19]. When the relationship between pixel and features values is too complicated, it is possible to implement the attack in two steps: first the attack which minimises the distortion in the feature domain is determined, then a new minimisation is carried out in the pixel domain, by adopting as objective function the distance between the attacked features and the feature vector resulting from the attack in the feature domain. An example of the above approach is given in [20], where a greedy approach is used to map back into the pixel domain a feature-domain attack against a machine-learning-based forgery detection method working in a higher-order feature domain. In many other papers, the authors focus on feature domain attacks without describing how the attack can be mapped back in the pixel or sample domain [10], [21], [22], [23].

Other problems associated with the use a gradient-descent image counter-forensics, include the necessity to preserve the integer nature of image pixels, which prevents the possibility to finely tune the magnitude of the descent step carried out at each iteration, and the highly irregular, non-differentiable, relationship between the pixels and the feature values. This is the case with the features proposed

WIFS'2017, December, 4-7, 2017, Rennes, France. 978-1-5090-6769-5/17/$31.00 ©2017 IEEE.

in [24], [25], [26], which are widely used in image forensic (see for instance [11], [27]). The quantisation of pixel residuals preceding the construction of the co-occurrence matrices at the heart of feature computation, in fact, makes the application of the gradient descent algorithm problematic.

In this paper, we present a gradient-based attack against SVM-based forensic techniques relying on high-dimensional SPAM features [24]. The attack works directly in the pixel domain even if the relationship between pixel values and SPAM features can not be inverted, and relies on a fast estimation of the gradient of the SVM output with respect to pixel values. Due to the necessity of constraining the pixel values to be integer numbers, the actual implementation of the attack departs from the classical gradient-descent algorithm, since the magnitude of the descent step is controlled by adjusting the number of pixels affected by the attack rather than by diminishing the modification undergone by each pixel. Our approach marks an improvement with respect to [20] where a minimum distortion attack is applied only for the case of linear SVM, when the point of the boundary which minimizes the distortion in the feature domain can be found analytically. For the more general case of non-linear SVM, the best target point cannot be identified, and the attack is carried out by randomly modifying some pixels of the image, in a way which reduces the SVM output, until the decision of the detector is reverted. Such an approach however departs from a minimum distortion one and, in the case of high dimensionality of the feature space, requires a large number modifications which lead to a unacceptable degradation of the quality of the attacked image. We tested the proposed attack against SVM detection of histogram stretching, adaptive histogram equalisation and median filtering. In all cases the attack succeeded in inducing a decision error with a very limited distortion, the PSNR between the original and the attacked images ranging from 50 to 70 dBs. The attack is also effective in a LK case, when a surrogate detector is attacked.

The rest of this paper is organised as follows: in Section II we state the general detection problem and characterize the behavior of the attack; then, in Section III we describe the gradient-based attack scheme against SVM detectors and the implementation procedure to reduce the computational complexity. The performance of the proposed attack are evaluated in Section IV by considering two global manipulations: the contrast-enhancement and the median-filtering.

II. Problem statement

We consider the problem of detection of global image manipulations, in which a forensic analyst searches an image for the traces left by a specific processing operator (e.g., resizing, filtering, contrast-enhancement, double JPEG compression). We let H_0 correspond to the case of manipulation absence and H_1 to the case in which the image has manipulated with a given global operator. Then, given an image I and a feature vector $\mathbf{f} = \mathbf{f}(I)$ extracted from the image ($\mathbf{f} \in \mathcal{F}$), the analyst applies a binary decision function ϕ such that $\phi(\mathbf{f}) = -1$ if the manipulation is not revealed (decision is in favor of H_0), $\phi(\mathbf{f}) = +1$ otherwise.The feature vector is a result of a dimensionality reduction. Many powerful tampering detectors proposed in the recent literature, e.g. [11], [27], exploit higher-order statistics and a quite large number of features so to capture many different types of dependencies among neighbouring pixels. Then, they resort to machine learning to decide between H_0 and H_1. Let y denote the decision labels, $y \in \{-1, 1\}$. The machine learning classifier ϕ is trained on a dataset $\mathcal{D} = \{\mathbf{f}_i, y_i\}_{i=1}^n$. The labels $y = \phi(\mathbf{f})$ are obtained by thresholding the learned discriminant function $g : \mathcal{F} \mapsto \mathbb{R}$; w.l.o.g., we assume that $\phi(\mathbf{f}) = +1$ if $g(\mathbf{f}) > 0$, -1 otherwise. By following a common trend in the literature, in this paper, we consider the Substractive Pixel Adjancency Model (SPAM) model for the features. Such a model is rich enough to perform an accurate classification (i.e., to capture the artifacts introduced by the various manipulations) and, at the same time, has a dimensionality which makes the use of an SVM still viable[1].

A. Attack model

Regarding the model for the attacker, his goal, knowledge and capabilities are defined as follow.

1) Attacker's goal: We assume that the attacker wants to modify a manipulated image in such a way that it is misclassified by the detector as a non-manipulated one. That is, he is interested in causing a false negative event (decision in favour of H_0 when H_1 holds). In doing so, the attacker wants to introduce the minimum distortion allowing to cross the decision boundary. This scenario is similar to those considered in [22], [23], [28], with the noticeable difference that in our case the attack is pursued in the pixel domain rather than in the feature domain[2]. In formula, given an image I, the goal of the attacker is to find an image I' such that $\phi(\mathbf{f}(I')) = -1$, and the distortion between I and I' is minimum.

2) Attacker's knowledge: In the PK scenario, the attacker has a perfect knowledge of the detector and then can build the attack by relying on the knowledge of ϕ, whereas in the LK scenario he does not know the training data $\mathcal{D}$. Then, he considers an approximation $\hat{\phi}$ of ϕ, built starting from a surrogate dataset.

3) Attacker's capability: We assume that the attacker can only modify the test data and not the training data.

III. A gradient-based attack to SVM-detectors

According to the problem statement given in the previous section, given a manipulated image I, the goal of the attacker is to solve the following problem:

$$I^* = \underset{I' : g(\mathbf{f}(I')) \leq -\nu}{\arg\min} \; d(I, I') \tag{1}$$

where $d(\cdot, \cdot)$ is a suitable distortion measure, e.g. the Mean Square Error (MSE). We observe that the admissibility region corresponds to the acceptance region of the test (decision in favor of H_0), with a safety margin ν. A larger ν means that the attacked image I^* will lie more inside the acceptance region, thus being more robust to perturbations of the decision boundary, at the price of a larger distortion. We take ν arbitrarily small in the PK scenario, whereas a larger ν is set for the LK case. For an SVM-based detector, the discriminant function of the classifier after training can be expressed as

$$g(\mathbf{f}) = \sum_{i=1}^{n} y_i \alpha_i k(\mathbf{f}, \mathbf{f}_i) - \rho \tag{2}$$

where n is the number of training samples, $k(\cdot, \cdot)$ is the kernel function, α is a vector of scalars (multipliers) and ρ is a bias term. In our experiments, we considered the RBF kernel function, for which $k(\mathbf{f}, \mathbf{f}_i) = \exp\{-\gamma ||\mathbf{f} - \mathbf{f}_i||_2^2\}$, where $\gamma > 0$ is a parameter of the kernel.

When d corresponds to the MSE distortion, solving (1) corresponds to search for the shortest path to the decision boundary. In principle, this can be done by applying the gradient descent algorithm. As discussed in Section I, however, the integer nature of the pixel values

[1] We notice that, although we focus on SVMs, in principle, the proposed attack can be applied to any classifier.

[2] We do not consider the fact that an adversary-aware defender may thwart the attack by moving the decision boundary (i.e., adjusting the threshold). To avoid this, the attacker might choose an I' which is misclassified with a higher confidence. This scenario would lead to a cat and mouse loop between the analyst and the attacker whose study is beyond the scope of this paper.

and the complicate form of $g()$ as a function of I, prevents the direct application of the gradient descent algorithm. Therefore, we propose a suboptimum, yet effective, iterative approach to solve (1), inspired to the gradient-descent method. The algorithm works as follow: first, an approximation of the gradient is derived to estimate the descent direction; then, the step-size for the descent is determined by properly choosing the percentage of to-be-modified pixels. If the decision boundary cannot be crossed by modifying at most a fraction 0.2 of the total number of pixels, then the modification is applied, the gradient is estimated again and the process is iterated[3]. In the following, we describe more in detail the proposed algorithm.

We start by estimating the gradient of $g(\mathbf{f}(I))$ with respect to I. Let δ be a small increment applied to a pixel position. The gradient ∇g in position (i,j) is given by:

$$(\nabla g)_{ij} = \frac{g(\mathbf{f}(I+\Delta_{ij})) - g(\mathbf{f}(I))}{\delta}, \qquad (3)$$

where Δ_{ij} is a matrix of the same size of I with only one non-zero entry, of value δ, in position (i,j). By definition, in order to compute the gradient, we should let $\delta \to 0$. In our case, due to the integer nature of pixel values, δ must be an integer, hence we let $\delta = 1$, thus obtaining only a rough approximation of the gradient.

The gradient descent algorithm should now proceed by modifying the image by a step of size ε along the direction specified by ∇g. The exact value of ε is a critical parameter, since small values result in a better precision, at the price of a slower convergence. In addition, with small values of ε, it is more probable that the descent is stuck into a local minimum. In the specific case studied here, there are some additional problems. First of all, values of ε for which pixels are modified by an amount smaller than one do not make sense since image pixels can take only integer values. A second problem, is related to the particular nature of the SPAM features [24]. Such features are computed by first computing image residuals based on simple predictors, and then computing a number of cooccurrence matrices on a heavily truncated version of the residuals. Truncation makes the relationship between pixel and feature values highly irregular and non-differentiable, thus making difficult to predict the effect of a perturbation of image pixels on the feature values (and hence on the SVM output). As a last problem, indirectly related to the impossibility of adopting very small values of δ and to the irregular dependence of the SVM output on pixel values, occurs when the number of modified pixels is too large, thus resulting in modifications of many neighbouring pixels. The effects that the modifications of these pixels have in the feature space are not independent (actually, they can be highly correlated), as a consequence, the effect of a joint modification is difficult to predict by looking at the gradient only. For all these reasons, there is no guarantee that gradient direction derived in (3) provides the steepest direction for the attack.

Our solution to the above problems consists in adjusting the strength of the attack by controlling the number of modified pixels rather than the amount of modifications undergone by each pixel and by keeping the number of modified pixels at each iteration below a certain percentage so to avoid that neighbouring pixels are modified too often. To be specific, let K denote the fraction of modified pixel and ε_K denote the step-size. The generic iteration of the attack in the pixel domain is defined by:

$$I' = I - \text{trunc}_1(\text{round}(\varepsilon_K \cdot \nabla g)), \qquad (4)$$

where $\text{trunc}_1()$ denotes the truncation to 1 and ε_K is chosen in such a way that the $l \cdot K$ pixels of the image with the larger intensity of the gradient are modified (where l is the total number of pixels). Accordingly, $\text{trunc}_1(\text{round}(\varepsilon \cdot \nabla g)$ is a matrix with lK entries ± 1, and the remaining entries equal to zero.

The choice of K is performed by starting from 0 and iteratively increasing the percentage of a small amount S until one of the following conditions is verified: i) $g(\mathbf{f}(I')) \leq -\nu$ (see equation (1)); ii) the maximum value of K is reached (set to 0.2). In the former case, a bisection method is applied to refine the value of K, so to find the smallest value for which $g(\mathbf{f}(I')) \leq -\nu$ (thus actually minimising the MSE); then, the attack in (4) is implemented with the corresponding ε_K. In the latter case, the attack is applied by using the value of K in the [0, 0.2] range yielding the minimum value of the decision function. The attack is then iterated by estimating the gradient on I' and so on. The main steps of the algorithm are detailed in Algorithms 1 and 2.

Algorithm 1 Gradient based attack

```
**Input**: $I$, original image; $\delta = 1$, pixel modification amount;
$g(\cdot)$, trained SVM classifier. **Output**: $I^*$, attacked image.
repeat
  Loop at each of locations $(i,j)$
    $I'_{ij} = I_{ij} + \delta$
    $(\nabla g)_{ij} = g(\mathbf{f}(I')) - g(\mathbf{f}(I))$
  end
  Select $K$ according to Algorithm2
  $\varepsilon_K \rightarrow (1-K)$-th percentile of $\nabla g$
  $I^* \leftarrow I - \text{trunc}_1(\text{round}(\varepsilon_K \cdot \nabla g))$
  $I = I^*$
until $g(\mathbf{f}) \leq -\nu$
return: $I^*$
```

Algorithm 2: search of K

```
Input: $\nabla g$, the gradient matrix; $S = 0.002$, the search step;
$I$, the input image, $s = 1$, $K(s) = 0$.
repeat
  $s = s + 1$
  $K(s) \leftarrow K(s-1) - S$
  $I_s \leftarrow$ attack $I$ according to (4) with $K(s)$
until g$(\mathbf{f}(I_s)) \leq -\nu$ or $K(s) > 0.2$
if $g(\mathbf{f}(I_s)) \leq -\nu$
  $K \leftarrow$ run bisection between $K(s)$ and $K(s-1)$
end
if $K(s) > 0.2$
  $K \leftarrow K(s^*)$, $s^*$ such that g$(\mathbf{f}(I^*_s))$ is minimum, $s^* \in [1 : s-1]$
end
return: $K$
```

In order to characterize the stopping condition $g(\mathbf{f}(I)) < -\nu$, in practice, we may alternatively look at the soft output of the SVM, namely $\hat{p}_1(\mathbf{f}(I))$, which provides an estimation of the probability that I belongs to the H_1 class. The decision boundary is crossed when $\hat{p}_1(\mathbf{f})$ is equal to 0.5. Then, the stopping condition can be rephrased as $\hat{p}_1(\mathbf{f}) < 0.5 - p_\nu$, where p_ν is a probability margin. Given that p_ν always ranges between 0 and 0.5, in the experiments, we found it easier to set this margin to determine how much the attack goes inside the acceptance region.

A. Reducing the complexity of the attack

A problem with the basic algorithm described so far is computational complexity. In fact, computing ∇g requires to evaluate the output of the SVM and the features the SVM relies on after each pixel modification. This results in a prohibitively high complexity, mostly due to the necessity of recomputing the features for each component of the gradient. To alleviate the computational burden, we can exploit dynamic programming to avoid recomputing the features from scratch for each pixel modification. This approach, is particularly easy and effective in the case of the SPAM features considered in this paper.

[3] We verified experimentally that it is more convenient to run more iterations rather than raising the percentage of pixels modified at each step.

The basic idea is that, since with the SPAM model each pixel modification results in the modification of a small number of features, equation (3) can be conveniently rephrased as follows:

$$(\nabla g)_{ij} = g(\mathbf{f} + \mathbf{v}_{i,j}(\delta)) - g(\mathbf{f}), \tag{5}$$

where $\mathbf{v}_{i,j}(\delta)$ denotes the impact on the feature vector of the modification of pixel (i, j). In the following we describe the exact procedure we used to compute $\mathbf{v}_{i,j}$. To start with, we observe that the calculation of SPAM features starts from the computation of the difference 2-D array for horizontal, vertical and diagonal directions. Then, these residuals are truncated with a given T and the co-occurrences are computed, where the value of T and the order of the co-occurrences depend on the specific order of the SPAM features considered. Specifically, given an $M \times N$ image I, the horizontal residual in the right-to-left direction is computed as $\overleftarrow{D}_{i,j} = \text{trunc}_T(I_{i,j+1} - I_{i,j})$, $1 \leq i \leq M, 1 \leq j \leq N - 1$. According to this extraction process, it is easy to argue that one pixel modification affects only two elements of the residual array for each direction. In fact, when the pixel in position (i, j) is modified by adding δ, residuals $\overleftarrow{D}_{i,j}$ and $\overleftarrow{D}_{i,j-1}$ are modified. Let us denote with $\overleftarrow{D'}_{i,j}$ and $\overleftarrow{D'}_{i,j-1}$ the modified values. Then, for the case of first-order SPAM features, where second-order co-occurrences are considered, as a consequence of one pixel modification, at most 6 elements of the co-occurrence matrix C are altered, corresponding to positions $(\overleftarrow{D}_{i,j-2}, \overleftarrow{D}_{i,j-1})$, $(\overleftarrow{D}_{i,j-1}, \overleftarrow{D}_{i,j})$ and $(\overleftarrow{D}_{i,j}, \overleftarrow{D}_{i,j+1})$, which are decreased by 1, and $(\overleftarrow{D}_{i,j-2}, \overleftarrow{D'}_{i,j-1})$, $(\overleftarrow{D'}_{i,j-1}, \overleftarrow{D'}_{i,j})$ and $(\overleftarrow{D'}_{i,j}, \overleftarrow{D}_{i,j+1})$, which are increased by 1. The situation is illustrated in Figure 1 where $\delta = 1$ is added to the entry in position $(3, 4)$ and $\overleftarrow{D}_{i,j-2}$, $\overleftarrow{D}_{i,j-1}$, $\overleftarrow{D}_{i,j}$ and $\overleftarrow{D}_{i,j+1}$ are denoted with u_0, u_1, u_2 and u_3.

In general, the computational complexity of the feature update step depends on the order of the SPAM model. In the case of m-th SPAM features, $(2 \cdot m + 4)$ elements of the co-occurence matrix of each directional residual must be updated. Then, with second-order SPAM features, 8 elements in the co-occurrence matrices change when one pixel is modified. This is the best performing SPAM model according to [24], where the truncation value is set to $T = 3$ and third order co-occurrences are considered for a 686 total dimensionality. In the rest of this paper, we will consider such SPAM$_{686}$ model. Our experiments show that the above procedure allows to reduce the complexity required to compute the feature vector by a factor of almost 50.

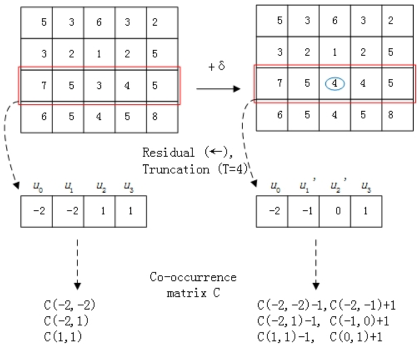


Fig. 1. Co-occurrences update for the horizontal residual (right to left) before and after one-pixel modification with 1st order SPAM.

IV. Experimental results

To evaluate the effectiveness of the attack described in the previous section, we focus on two specific image forensic problems: contrast enhancement and median filtering detection. For the case of contrast enhancement, we consider both the case of histogram stretching and adaptive histogram equalization, whereas for the case of median filtering detection, we consider various filtering windows (3×3, 5×5 and 7×7). We carried out our tests by considering 2000 uncompressed grayscale images from the RAISE dataset [29]. The manipulated images are then created by processing the images in the Matlab environment. For adaptive histogram equalization, we considered the contrast-limited variant (CLAHE) which prevents over-amplification of noise in homogeneous regions. In our experiments, we set the clip-limit parameter to 0.02. To build the detectors, both the original set and the manipulated set are split as follows: 1600 images are used for training the SVM, the remaining 600 for testing. We denote with HS the set of the histogram stretched images, with AHE the set of the contrast-limited adaptive histogram equalized and with $MF3$, $MF5$ and $MF7$ the set of images processed with median filtering, with filtering window 3×3, 5×5 and 7×7 respectively. The SVM model is built by using the tools provided by the LIBSVM library.

A. PK case

Assuming that the attacker has full knowledge of the trained SVM, we implement the attack described in the previous section on the 600 manipulated images of the testing set. Note that in the PK scenario, it is reasonable to assume that the safety margin is arbitrarily small, and then the attacker is satisfied as soon as the soft output of the SVM falls below 0.5, that is $\hat{p}_1(\mathbf{f}) < 0.5$. The average number of iterations for the attack are: 0.965 for HS, 2.75 for AHE, 1.22 for $MF3$, 1.47 for $MF5$ and 1.83 for $MF7$. The fact that the average number of iterations is below 1 for the attack to the histogram stretching detector can be explained by noticing that there is a fraction of manipulated images for which the detector fails even without attack (see Table II) The average fraction of pixels modified by the attack is reported in Table I. We can see that, in order to successfully attack the median filtering detector, a large number of pixels of the image must be changed. Expectedly, this number increases when the manipulation the attack wants to conceal is stronger, as it is the case of filtering with larger windows. An example of manipulated image before and after attack is provided in Figure 2. The soft output of the SVM is reduced from 0.862 to 0.491 in the case of attack to the histogram stretching detector, from 0.998 to 0.484 for the attack to CLAHE detector, and from 0.999 to 0.489 for the median filtering detector with 3×3 window. The visual quality of the attacked images is good; specifically, the PSNR between attacked and manipulated image in the three cases is 82 dB, 60.9 dB and 56.5 dB respectively.

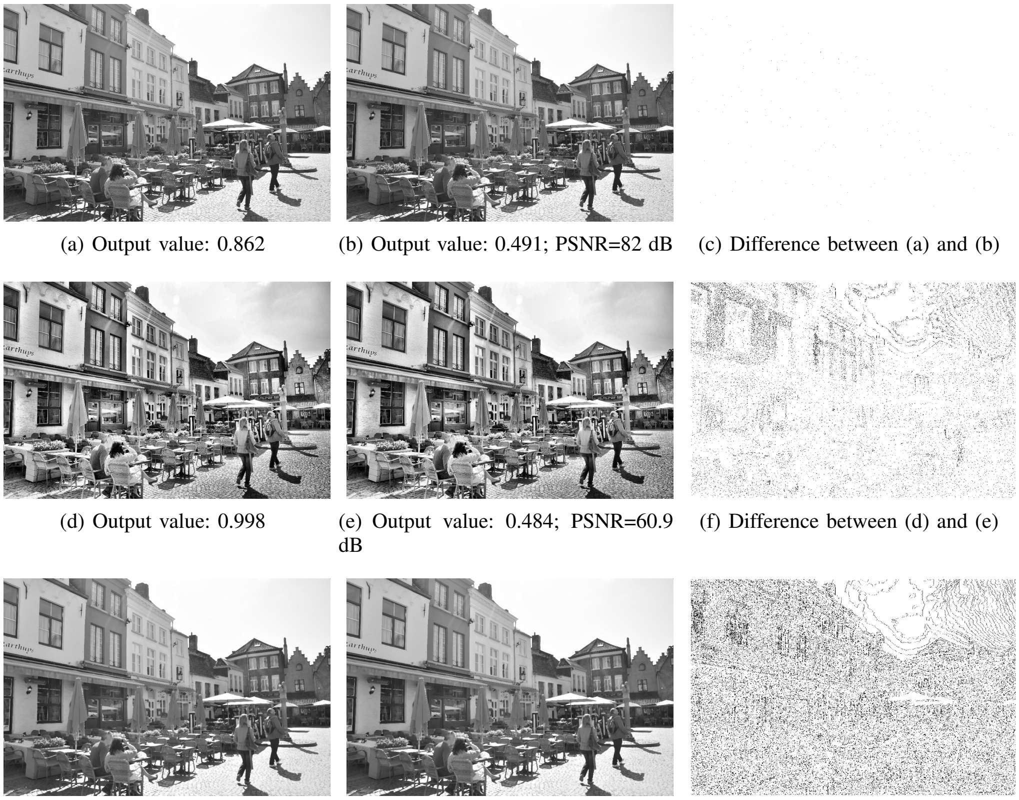
TABLE I
Average fraction of pixels modified by the attack

	HS	AHE	MF3	MF5	MF7
Mean	0.0052	0.138	0.1155	0.1365	0.1553
Std. Dev.	0.0084	0.0549	0.0473	0.0561	0.0609

The average performance of the attack are evaluated in terms of visual quality and capability of inducing a decision error. The success rate of the attack is reported in Table II, where the percentage of misclassified images is reported for the various cases. The PSNR and the Structural Similarity (SSIM) index [30] have been used to evaluate the visual quality of the attacked images. As we can see from the results are shown in Table III, the attacked images have a very good quality.

B. LK case

We also assessed the performance of the proposed attack in a case of limited knowledge, when the attacker knows the feature selection process and the type of classifier adopted by the detector, but not



(a) Output value: 0.862

(b) Output value: 0.491; PSNR=82 dB

(c) Difference between (a) and (b)

(d) Output value: 0.998

(e) Output value: 0.484; PSNR=60.9 dB

(f) Difference between (d) and (e)

(g) Output value: 0.999

(h) Output value: 0.489; PSNR=56.5 dB

(i) Difference between (g) and (h)

Fig. 2. (a) An histogram stretched image I; (b) the attacked version; (c) the enhanced difference between the two (better viewed on screen). (d) The adaptive histogram equalized version of I; (e) the attacked adaptive histogram equalized version; (f) the enhanced difference. (h) The median filtered version of I, (i) the attacked median filtered version and (j) the enhanced difference.

TABLE II
ERROR PROBABILITY OF THE DETECTORS

	HS	**AHE**	**MF3**	**MF5**	**MF7**
Manipulated	2.83%	0	0	0	0
Attacked	99.83%	100%	100%	99	98.7

TABLE III
QUALITATIVE PERFORMANCE OF THE GRADIENT-BASED ATTACK

	Mean PSNR	**Mean SSIM**	**Std. dev. SSIM**
HS	74.3 dB	0.99997	0.00006
AHE	55 dB	0.9993	0.00049
MF3	57.7 dB	0.9992	0.00036
MF5	56.8 dB	0.9987	0.00075
MF7	56 dB	0.9983	0.0011

the training set. In this case, he cannot build an exact replica of the detector ϕ. Then, he builds its own version of the detector by training the SVM on a different set of images and uses such surrogate detector $\hat{\phi}$ to run the gradient-based attack. In our experiments, we built $\hat{\phi}$ by training the SVM on a different set of 1600 images from the RAISE dataset. Table IV shows the performance of the attack in the LK case against the histogram stretching detector. Results are reported for the case in which the attacker is satisfied with any point that crosses the boundary (as for the PK case), named *Attacked*, and the case in which a margin p_ν equal to 0.2 and 0.4 is used, and then, the stopping condition for the attack is $\hat{p}_1(\mathbf{f}) \leq 0.3$ and $\hat{p}_1(\mathbf{f}) \leq 0.1$, which are named *Attacked-m02* and *Attacked-m04* respectively. The error probability in the first column, named $P_e(\hat{\phi})$, refers to the result of the test on the surrogate dataset $\hat{\phi}$. The *true* error probability, that is the error probability of the detector ϕ named $P_e(\phi)$, is reported in the second column. We see that, because of dataset mismatch[4], the performance of the attack decreases with respect to the PK case. When the safety margin is increased the attack is more powerful, being more robust to perturbations of the decision boundary, as the one caused by the use of a different dataset for training. Expectedly, this goes at the expense of a slightly larger distortion.

TABLE IV
ERROR PROBABILITY OF THE HS DETECTOR (LK CASE)

	$\mathbf{P_e}(\hat{\phi})$	$\mathbf{P_e}(\phi)$	**Mean SSIM**	**Mean PSNR**
Attacked	100%	53%	0.99996	73.9766
Attacked-m02	100%	80.5%	0.99995	72.6223
Attacked-m04	100%	100%	0.99994	71.2038

V. CONCLUSION

We have developed and tested a general attacking procedure on image forensic detectors based on the use of a large dimensionality feature vector coupled with machine learning, which works directly in the pixel domain. We have specialised the attack to work efficiently against an SVM detector relying on SPAM features, nevertheless the general procedure can be applied to other detectors relying on different feature models, e.g., the Spatial Rich Model (SRM) [26],

[4] Since we are considering the same database, strictly speaking, this is not a case of database mismatch, whose investigation is left as future work.

possibly at the expenses of a higher complexity. It is worth saying that, although we focus on uncompressed images, our method can be also applied when JPEG images are considered. Since the compression tends to erase the modifications introduced in the pixel domain, the attacker has to get more inside the detection region (so that it remains inside after quantization); alternatively, we could directly apply our gradient-inspired attack to the quantized DCT coefficients[5]. Interesting extensions include tests on a wider class of detectors, and studying the impact of a detector mismatch in a fixed feature space (e.g. an neural network detector attacked with a surrogate SVM-based detector). The good performance of the proposed attack poses new challenges to the image forensic community. Some possible approaches to counter the attack include the resort to fusion of several forensic tools operating in different feature domains, and the training of an adversary-aware version of the detector that is able to recognise the possible traces left by the gradient-based attack.

Acknowledgment

This work has been partially supported by a research sponsored by DARPA and Air Force Research Laboratory (AFRL) under agreement number FA8750-16-2-0173, by the National Key Research and Development Program of China (2016YFB0800404), National Natural Science Foundation of China (61332012, 61672090, 61572052), Fundamental Research Funds for the Central Universities (2015JBZ002). The U.S. Government is authorised to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA and Air Force Research Laboratory (AFRL) or the U.S. Government. We thank Luisa Verdoliva and Francesco Marra from the University of Napoli for their help in comparing our method to the one in [20].

References

[1] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme, "Can we trust digital image forensics ?" in *ACM Multimedia 2007, Augsburg, Germany*, September 2007, pp. 78–86.

[2] M. Barni and F. Pérez-González, "Coping with the enemy: advances in adversary-aware signal processing," in *ICASSP 2013, IEEE Int. Conf. Acoustics, Speech and Signal Processing*, Vancouver, Canada, 26-31 May 2013, pp. 8682–8686.

[3] M. Barni and B. Tondi, "Source distinguishability under distortion-limited attack: An optimal transport perspective," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 10, pp. 2145–2159, 2016.

[4] M. Kirchner and R. Bohme, "Hiding traces of resampling in digital images," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 582–292, December 2008.

[5] M. C. Stamm and K. J. R. Liu, "Anti-forensics of digital image compression," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 50–65, September 2011.

[6] P. Sutthiwan and Y. Q. Shi, "Anti-forensics of double JPEG compression detection," in *Shi Y.Q., Kim HJ., Perez-Gonzalez F. (eds) Digital Forensics and Watermarking. IWDW 2011. Lecture Notes in Computer Science, vol 7128*. Springer, 2012.

[7] M. Fontani and M. Barni, "Hiding traces of median filtering in digital images," in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*. IEEE, 2012, pp. 1239–1243.

[8] Z.-H. Wu, M. C. Stamm, and K. R. Liu, "Anti-forensics of median filtering," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3043–3047.

[9] W. Fan, K. Wang, F. Cayre, and Z. Xiong, "Median filtered image quality enhancement and anti-forensics via variational deconvolution," *IEEE transactions on information forensics and security*, vol. 10, no. 5, pp. 1076–1091, 2015.

[10] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Proc. of Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 387–402.

[11] X. Qiu, H. Li, W. Luo, and J. Huang, "A universal image forensic strategy based on steganalytic model," in *Proceedings of IH&MMSec'14. 2nd ACM workshop on Information hiding and multimedia security*, Salzburg, Austria, 11-13 June 2014.

[12] J. Kodovskỳ and J. Fridrich, "Steganalysis in high dimensions: Fusing classifiers built on random subspaces," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2011, pp. 78 800L–78 800L.

[13] M. Barni, M. Fontani, and B. Tondi, "Universal counterforensics of multiple compressed jpeg images," in *Digital-Forensics and Watermarking*. Springer, 2014, pp. 31–46.

[14] ——, "A universal technique to hide traces of histogram-based image manipulations," in *Proc. of the ACM Multimedia and Security Workshop*, Coventry, UK, 6-7 September 2012, pp. 97–104.

[15] M. Barni, M.Fontani, and B. Tondi, "A universal attack against histogram-based image forensics," *International Journal of Digital Crime and Forensics (IJDCF)*, vol. 5, no. 3, 2013.

[16] P. Comesaña and F. Perez-Gonzalez, "The optimal attack to histogram-based forensic detectors is simple(x)," in *Information Forensics and Security (WIFS), 2014 IEEE International Workshop on*, Dec 2014, pp. 137–142.

[17] C. Villani, *Optimal Transport: Old and New*. Berlin: Springer-Verlag, 2009.

[18] B. Tondi, *Theoretical Foundations of Adversarial Detection and Applications to Multimedia Forensics*. University of Siena: PhD Thesis, September 2016.

[19] X. Chu, M. C. Stamm, Y. Chen, and K. R. Liu, "On antiforensic concealability with rate-distortion tradeoff," *IEEE Transactions on Image Processing*, vol. 24, no. 3, pp. 1087–1100, 2015.

[20] F. Marra, G. Poggi, F. Roli, C. Sansone, and L. Verdoliva, "Counter-forensics in machine learning based forgery detection," in *SPIE/IS&T Electronic Imaging*. International Society for Optics and Photonics, 2015, pp. 94 090L–94 090L.

[21] C. Pasquini, P. Comesaña-Alfaro, F. Pérez-González, and G. Boato, "Transportation-theoretic image counterforensics to first significant digit histogram forensics," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 2699–2703.

[22] N. Dalvi, P. Domingos, P. Mausam, S. Sanghai, and D. Verma, "Adversarial classification," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004, pp. 99–108.

[23] B. Nelson, B. I. Rubinstein, L. Huang, A. D. Joseph, S. J. Lee, S. Rao, and J. Tygar, "Query strategies for evading convex-inducing classifiers," *Journal of Machine Learning Research*, vol. 13, no. May, pp. 1293–1332, 2012.

[24] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on information Forensics and Security*, vol. 5, no. 2, pp. 215–224, 2010.

[25] T. Pevny and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis," *Proc. SPIE*, vol. 6505, pp. 650 503–650 503–13, 2007.

[26] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868–882, 2012.

[27] D. Cozzolino, G. Poggi, and L. Verdoliva, "Splicebuster: A new blind image splicing detector," in *Information Forensics and Security (WIFS), 2015 IEEE International Workshop on*. IEEE, 2015, pp. 1–6.

[28] D. Lowd and C. Meek, "Adversarial learning," in *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, 2005, pp. 641–647.

[29] D.-T. Dang-Nguyen, C. Pasquini, V. Conotter, and G. Boato, "Raise: A raw images dataset for digital image forensics," in *Proceedings of the 6th ACM Multimedia Systems Conference*, ser. MMSys '15. New York, NY, USA: ACM, 2015, pp. 219–224. [Online]. Available: http://doi.acm.org/10.1145/2713168.2713194

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[5] Needless to say that this goes at the cost of higher computational time, due to the necessity to evaluate the DCT and I-DCT at each iteration.