Open Set Source Camera Attribution and Device Linking

Filipe de O. Costa[a], Ewerton Silva[a], Michael Eckmann[b], Walter J. Scheirer[c], Anderson Rocha[a]

[a]*Institute of Computing, University of Campinas (Unicamp)*
Av. Albert Einstein, 1251, Cidade Universitária "Zeferino Vaz",
Campinas, SP, Brazil – CEP 13083-852
Phone +55 19 3521-5854 – Fax +55 19 3521-5838

[b]*Dept. of Mathematics and Computer Science,*
Skidmore College, Saratoga Springs, NY, USA – ZIP Code 12866

[c]*Dept. of Molecular and Cellular Biology, School of Engineering and Applied Sciences, Center for Brain Science,*
Harvard University, 52 Oxford Street, Cambridge, MA, USA – ZIP Code 02138

Abstract

Camera attribution approaches in digital image forensics have most often been evaluated in a closed set context, whereby all devices are known during training and testing time. However, in a real investigation, we must assume that innocuous images from unknown devices will be recovered, which we would like to remove from the pool of evidence. In pattern recognition, this corresponds to what is known as the *open set recognition problem*. This article introduces new algorithms for open set modes of image source attribution (identifying whether or not an image was captured by a specific digital camera) and device linking (identifying whether or not a pair of images was acquired from the same digital camera without the need for physical access to the device). Both algorithms rely on a new multi-region feature generation strategy, which serves as a projection space for the class of interest and emphasizes its properties, and on *decision boundary carving*, a novel method that models the decision space of a trained SVM classifier by taking advantage of a few known cameras to adjust the decision boundaries to decrease false matches from unknown classes. Experiments including thousands of unconstrained images collected from the web show a significant advantage for our approaches over the most competitive prior work.

Keywords: Open Set Recognition; Camera Attribution; Device Linking; Decision Boundary Carving.

1. Introduction

With the rise of digital photography, a growing number of digital images have become associated with evidentiary pools for criminal and civil proceedings. This presents an often frustrating dilemma for those charged with verifying the integrity and authenticity of such images, since they are not always generated by known devices, and can be modified with ease [4]. Moreover, with an estimated 250 million images being added to Facebook every day[1] from an enormous set of unknown sources, looking for images from a particular camera of interest becomes a significant challenge. In this article, we investigate a fundamentally new approach for the specific problems of *Image Source Attribution* and *Device Linking* in the context of open set recognition, where not all cameras are known during training time (Fig. 1).

Similar to a ballistics exam in which bullet scratches allow forensic examiners to match a bullet to a particular gun [5], image source attribution techniques look for artifacts left in an image by the source camera such as dust on the lens, the interaction between device components and the light, factory defects, and other effects [6]. Sensor attribution problems span a variety of devices such as cameras [7, 8, 1, 2], printers [9, 10], and scanners [11, 12]. Beyond a basic examination of the EXIF headers, which contain textual information about the digital camera type and the conditions under

Email addresses: filipe.costa@ic.unicamp.br (Filipe de O. Costa), ewerton.silva@ic.unicamp.br (Ewerton Silva), meckmann@skidmore.edu (Michael Eckmann), wscheirer@fas.harvard.edu (Walter J. Scheirer), anderson.rocha@ic.unicamp.br (Anderson Rocha)

[1]Public disclosure to the U.S. Securities and Exchange Commission by Facebook: http://goo.gl/cAU0O

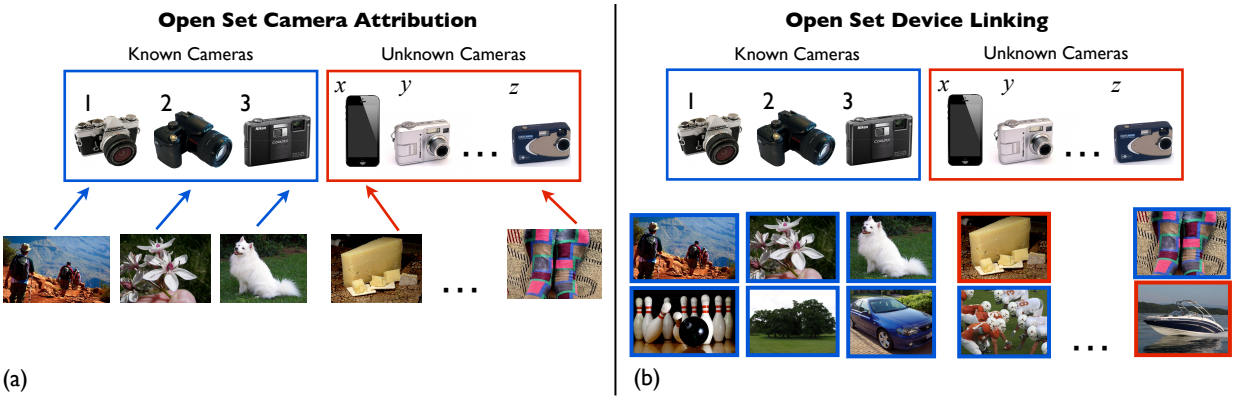

Figure 1: Image source camera attribution is the process of identifying whether or not an image was captured by a specific digital camera. Device linking is the process of identifying whether or not a pair of images comes from the same digital camera – without the need for physical access to the device. While much progress has been made in both areas, the most promising recent approaches [1, 2, 3] restrict evaluation to a closed set scenario, where all cameras are known during training and testing. For instance, closed set camera attribution considers only images from known cameras during training and testing (blue cases in (a)), while closed set device linking considers matched and non-matched pairs of images from known cameras (blue cases in (b)). A more realistic scenario for real world investigations is open set evaluation, where during testing (the operational scenario) we must consider images from unknown cameras. For camera attribution, images from unknown cameras should be rejected to avoid false attribution (*e.g.* the red arrow cases in (a)). Similarly, pairs of images containing images from unknown cameras should also be rejected to avoid false linking (*e.g.* the red/blue and blue/red pairs on the right of (b)).

which the photograph was taken but can be easily tampered with or destroyed [4], a class of methods exists that identifies the brand/model of the source camera [13, 14] by directly considering the image data. These methods generally perform an analysis of color interpolation algorithms. However, many camera brands use components by only a few factories, and the color interpolation algorithm is the same (or very similar) among different models of the same brand of cameras [4, 6].

Since fine-grained categorization is of more value to the field of digital image forensics, most source attribution approaches have the objective of identifying the specific camera that took a photograph instead of just the device's brand and model. There is some previous work that analyzes device defects for image source identification [7, 15], as well as artifacts caused by dust on the lens at the time the image was taken [8]. The problem with such methods is that some current camera models do not contain any obvious defects, while others eliminate defective pixels by post-processing their images on-board. Further, some artifacts are strictly temporal by nature and can be easily destroyed (*e.g.*, the lens may be cleaned or switched). In response, forensic experts have given special attention to methods based on sensor pattern noise (SPN) because they can identify specific instances of the same camera model by using the deterministic component of SPN [1, 2]. This component is a robust fingerprint for identifying source cameras and verifying the integrity of images because it is the result of factors such as the variable sensitivity of each sensor element to light, the inhomogeneity of silicon wafers, and the uniqueness of manufacturing imperfections that even sensors of the same model possess [4, 6, 1].

SPN is also useful for cases in which all a forensic examiner has is a set of photographs and the question is to determine whether or not the photographs were taken by the same camera. This challenge is known in the literature as device linking. With device linking methods, we can attest that a set of images was taken by a specific camera by comparing each image to another image that we know belongs to the specific camera – without needing physical access to it. This is a practical problem with potentially important implications in the age of social media. With the possibility of different photo albums spread across sites (Flickr, Facebook, Picasa, etc.), useful evidence can be isolated if an investigator knows that certain suspect images came from the same device, even if she only has access to the public images, and not the camera itself. Solutions to this problem also apply to the scenario of discovering whether or not illegal photos posted on the Internet were generated by a known stolen camera (when an investigator is in possession of a collection of reference images). Further, the commercial space has also

expressed interest in the device linking problem: a premium service is already available for public and private investigators[2].

Nearly all of the prior work in image source attribution and device linking was evaluated in a closed set scenario, in which one assumes that an image under investigation was generated by one of n known cameras available during training. However, it is possible that the image may have been generated by an unknown device not available during training (*i.e.*, in the set of suspect devices under investigation). Therefore, it is essential to model attribution problems as *Open Set* scenarios (Fig. 1), which resemble a realistic situation where we only have partial knowledge of the world we are modeling. In this case, we need a classification model for the few available classes (cameras under investigation), while trying to take the large unknown set of unavailable cameras into consideration.

In this article we describe a new feature generation approach for open set classification, as well as a new method for adjusting the decision boundary of an SVM classifier, based on the available knowledge of the world during training, called *decision boundary carving* [16]. For image source attribution in an open set scenario, we obtain better results compared to state-of-the-art approaches for a very large dataset composed of 13,210 images from 400 different cameras, including "in the wild" images from 375 cameras taken from public Flickr albums. Similarly, we achieve higher accuracies for the device linking problem in an open set scenario for a dataset composed of 25,000 pairs images sampled from the same set used for attribution. Our approach can be used by investigators to analyze images with different resolutions and acquisition circumstances, with good classification results across all conditions. In addition, the classification methods we propose are general enough to also be useful in a diverse set of classification problems outside of the realm of forensics.

Our contributions in this article, which is an extension of our recent conference paper [16], can be summarized as follows:

1. A review of the recent literature on camera attribution problems in the context of realistic open set recognition scenarios.
2. A new feature generation approach that addresses the open set classification problem in digital image forensics by serving as a projection space for the class of interest.
3. Algorithms for image source attribution and device linking incorporating decision boundary carving – a new approach for modeling the decision space of a trained SVM.
4. Large scale open set experimentation incorporating thousands of unconstrained images from the web, including an assessment of statistical significance for all algorithms considered.

[2]Quintel Intelligence: `http://goo.gl/0pRN9`

2. Related Work

To expand upon what we have touched on above, the problem of matching an image to the device that captured it is known in the forensics literature as *image source attribution* [4]. There are several features one can rely on for tackling this problem such as environment, noise, dust on the lens, hardware and component imperfections, and effects of operational conditions. These same features can also be used to solve the related problem of *device linking*, where the objective is to verify whether or not two images come from the same camera without the need for physical access to the actual device.

2.1. Image Source Attribution

Recent approaches have explored sensor pattern noise (SPN) for solving the image source attribution problem. SPN has drawn special attention from the forensics community because of its ability to identify a specific camera and not just the brand/model of the device. In general, one can consider two types of sensor noise patterns: Fixed Pattern Noise (FPN) and Photo Response Non-Uniformity Noise (PRNU).

FPN is caused by dark currents (the result of the accumulation of electrons in each sensor element of the device due to thermal action [7]). Normally, it can be eliminated by some camera models on-the-fly. The PRNU, on the other hand, is divided into low-frequency defects noise (LFD) and pixel non-uniformity noise (PNU). LFD is mainly related to light refraction on particles near the camera, or by zoom configurations, and does not make a good candidate for forensics attribution because of its unstable nature. Conversely, PNU is more stable because it is caused by the interaction between the light and each sensor element of the sensor array, revealing important clues for forensics.

Lukáš et al. [1] have explored using PNU for the image source attribution problem. Given a set of images K generated by a camera C, they calculate the residual

noise R_{I_j} for each image $I_j \in K$ using a discrete wavelet transform based filter F

$$R_{I_j} = I_j - F(I_j) \tag{1}$$

Then, the method calculates a reference pattern SPN_c of the sensor pattern noise of the camera C by averaging all residual noise in the image set (Eq. 2). The residual noise is used in this step to reduce the influence of scene detail.

$$SPN_c = \frac{1}{k}\sum_{i=1}^{k} R_{I_i}, \text{where } k = |K|. \tag{2}$$

A correlation value ρ_c is calculated between the residual noise R_J of an image J under investigation and the SPN_c of a camera

$$corr = \rho_c(R_J, SPN_c) = \frac{(R_J - \overline{R}_J)\cdot(SPN_c - \overline{SPN}_c)}{\|R_J - \overline{R}_J\|\cdot\|SPN_c - \overline{SPN}_c\|}, \tag{3}$$

where the mean value of the pixels is denoted by the bar above a symbol. For deciding a match, a threshold τ is calculated using the Neyman-Pearson approach to minimize the false rejection rate (FRR) while imposing a bound on the false acceptance rate (FAR). A match between an image and camera C exists if the value of ρ_c is higher than τ. High accuracy rates were reported by Lukáš et al. [1] for a test of nine cameras, and the results were later confirmed by others [17, 18].

Extending Lukáš et al.'s method [1], Li [2] proposed a sensor pattern noise enhancement method to reduce the influence of the scene content in the noise component. Li argues that the high frequencies (*e.g.*, object edges) in an image directly affect its PRNU component, which subsequently affects the camera identification results. In Li's method, given one image $I_p \in K$, after extracting its noise $n = R_{I_p}$ according to Eq. 1, there is a normalization of each pixel $\eta(x, y)$, generating the enhanced noise $\eta_e(x, y)$. Eq. 4 represents the model with the best results ($\alpha = 7$).

$$\eta_e(x,y) = \begin{cases} e^{-0.5n^2(x,y)/\alpha^2}, & \text{if } 0 \le n^2(x,y); \\ -e^{-0.5n^2(x,y)/\alpha^2}, & \text{otherwise;} \end{cases} \tag{4}$$

Li reported lower false-positive rates than [1] for a scenario with six cameras and the center 512×512 region of interest of the image.

In addition to [1] and [2], several other related methods have been proposed in the literature such as those based on clustering image sets [19, 20], or approaches that combine image source attribution information and color filter interpolation features [21]. To complicate things, there are also counter-forensic techniques that focus on discovering and exploiting inconsistencies in camera identification methods to foil the attribution process [22, 23].

2.2. Device Linking

Parallel to the sensor attribution problem, Goljan et al. [3] proposed an approach for checking whether a pair of images was captured by the same acquisition device without having physical access to that device. The device linking algorithm proceeds as follows. Given a pair of images I_1 and I_2, the PNU component is extracted as in [1] (Eq. 1) and the images are directly compared by means of the Normalized Cross-Correlation (NCC)

$$\mathrm{NCC}(u,v) = \frac{\sum_{i,j}(R_{I_1}[i,j] - \overline{R}_{I_1})\cdot(R_{I_2}[i+u, j+v] - \overline{R}_{I_2})}{\sqrt{\sum_{i,j}(R_{I_1}[i,j] - \overline{R}_{I_1})^2}\cdot\sqrt{\sum_{i,j}(R_{I_2}[i,j] - \overline{R}_{I_2})^2}}. \tag{5}$$

where the bar above a symbol represents the mean.

The algorithm requires that both images be of the same size for comparison. For this reason, Goljan et al. recommend padding the images with zeros (when necessary) before calculating the NCC. Cropping two regions of the same size around the center of both images is also plausible [3]. For the decision step, Goljan et al. explore measures of peak sharpness on the NCC such as the ratio between the primary and the secondary peaks (PSR). A PSR that exceeds a threshold (established to minimize the false positive rate of misclassifying two images as coming from the same camera) is an indication that both images were captured by the same camera.

2.3. Limitations of the Prior Work in Sensor Attribution and Device Linking for Open Set Problems

Although the approaches [1, 2, 3] we have reviewed are effective, it is important to understand that they have some deficiencies. To estimate the threshold τ responsible for matching an image to a camera or for deciding that two images come from the same camera, it is assumed that one has images from all possible cameras, and has subsequently labeled the entire space in binary fashion as either positive (generated by the camera under investigation) or negative (otherwise). More specifically, when creating an algorithm for deciding whether an image matches its acquisition camera, one needs to eliminate false matches to other possible cameras found in the wild that are unknown during training. This is known as open set classification.

To highlight the problem with current attribution methods, consider the following example: suppose that investigators seize and retain several cameras and hard disks containing child pornography. In a closed set camera source attribution analysis for this case, the investigators would assume that the images come from one of the seized cameras. However, this assumption is naïve and would easily lead to unexpected false matches. Thus, open set attribution is the mode they should operate in. If we are strict about what an open set means, we can say that Lukáš et al. [1] and Li [2] partially dealt with it by defining specific reference patterns for each image, which aims at ruling out unknown devices. In this article, we go beyond a simple characterization of the problem and explicitly deal with it in the context of a novel machine learning approach.

Although important, the open set recognition problem has received limited attention in the pattern recognition literature thus far. For instance, in a study of face recognition evaluation methods proposed by Phillips et al. [24], the authors define a threshold τ such that all scores from an algorithm must necessarily exceed τ to be considered a match. However, being greater than τ is a necessary but not sufficient condition for avoiding false matches. Possible unknown impostors may exist (exceeding the threshold) since it is impossible to train the system with all possible impostors. Indeed, nearly all works that we could find in the literature claiming to address the open set problem do so by simply setting a threshold.

A more sophisticated "1-vs-Set Machine" algorithm based on the linear SVM [25] is described by Scheirer et al. [26]. To improve the overall open set recognition error, the 1-vs-Set Machine balances the unknown classes by obtaining a core margin around the decision boundary from a base SVM, specializing the resulting half-space by adding another plane and then generalizing or specializing the two planes to optimize empirical and open space risk. The process uses the open set training data and the risk model to define a new "open set margin". Our work here can be viewed as a variation of this approach that considers a non-linear kernel and moves only a single plane.

In the forensics literature, Wang et al. [27] perform open set camera model identification using features based on Color Filter Array (CFA) coefficients as proposed in [14, 13]. The authors use a combination of binary SVM classification approaches: Two-class SVMs (TC-SVM) and One-class SVMs (OC-SVM) [28]. More specifically, the OC-SVM may be considered a solution for open set problems, given that it is not restricted to a defined sampling of negatives. The authors use only two out of 17 available cameras for training (one class of interest and one for outlier definition, which can be seen as a form of accounting for the unknown) and all 17 cameras for testing. The work reported a true positive rate of approximately 91%. Two limitations of the method are that, considering CFA coefficients, one can identify only the brand/model of the camera that generated an image. Second, one-class solutions for open set problems tend to generalize poorly [29].

3. A Methodology for Open Set Camera Attribution and Device Linking

In this work, we do not assume that an image under investigation was generated by an available camera. The key difference from prior work is that for learning, we assume that we have access to some classes of interest (suspicious cameras under investigation), but a vastly undersampled representation of the space of negative classes (all other cameras in the world) when calculating the decision boundary from a rich source of features. Since the space of all negative classes is beyond our means to quantify, we solve for a more feasible objective during training. Fig. 2 depicts an example of this open set classification problem.

Our approaches for open set source camera attribution and device linking share two core elements:

1. Multi-Region Feature Characterization;
2. Open Set Classification.

3.1. Multi-Region Feature Characterization

Before performing any training or classification, we need to characterize the images under investigation and represent them in a form that is more suitable for computation. The first step is to determine a region of interest for characterization in each image. Lukáš et al. [1] proposed the use of the image's central region while Li [2] considered the whole image in some cases. Choosing a common region for all images (*e.g.*, the central region) may be better for image source camera attribution and device linking when we have images with different resolutions.

However, according to Li and Satta [30], different regions of the image can contain different information with respect to the acquisition process. Therefore, our method considers multiple regions of one image instead of just the central one. We calculate nine 512×512-pixel regions of interest (ROIs), as shown on the lefthand side of Fig. 3. Regions in the center of the image (1-5)

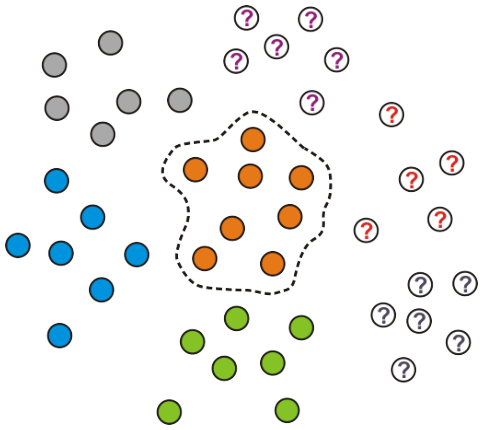

Figure 2: An example of open set classification. The above diagram shows a known class of interest ("orange"), surrounded by other classes that are not of interest, which can be known ("blue," "gray," "green"), or unknown ("?"). The "orange" class may refer to images of a suspicious camera under investigation and the "blue," "gray" and "green" classes may refer to images of three other known cameras. Since this is an open scenario, we can only use the information about the four known cameras for calculating a decision threshold. However, the real world is more complex, and there are countless cameras that cannot be considered as part of the negative class in training. Any algorithm used operationally must address this aspect of "unknowns" in the problem.

assume coincidence with the principal axis of the lens and are expected to have a great deal of scene detail, as amateur photographers usually focus the object of interest in the center of the lens. Peripheral regions (6-9) are also important because some cameras suffer from vignetting, which is a radial falloff of intensity from the center of the image, causing a reduction of an image's brightness or saturation at the periphery [30, 31]. These are interesting and useful properties for attribution and linking analysis.

Different from Lukáš et al. [1], who have considered only gray-scale images when calculating the PNU for camera attribution, we calculate the PNU (as defined in Eqs. 1 and 2) for each region of interest by considering the color channels R (red), G (green), and B (blue), as well as the Y channel (luminance, from YCbCr color space), which is a combination of the R, G and B channels (a gray-scale version of the image) [32]. We end up with 36 reference noise patterns to represent each camera, giving us a larger space of information to consider during learning. The regions and color spaces can be thought of as a projection space for the class of interest, emphasizing its properties. This is essential for reaching the levels of discriminability required for open set scenarios where any number of unknown images can be encountered. Further, this type of region characterization allows us to compare images with different resolutions without color interpolation artifacts, and it is not necessary to do zero-padding, for instance, when comparing images of different sizes. The feature vector representing an image is created by calculating the correlation between each ROI of the image and the corresponding noise pattern for each camera, according to Eq. 3. The final result is a 36-dimensional feature vector. Fig. 3 depicts this process.

For device linking, we collect feature vectors from the positive class of interest and from the known negative classes. In this case, a positive class of interest consists of actual examples of pairs of images coming from the same camera. A negative class consists of examples of pairs of images coming from different cameras. Since we do not have access to any physical device, we need to check if two images originate from the same camera based solely on their content and properties. We extract a feature vector for a pair of images using a 2-D correlation function (Eq. 6) between the correlated noises of each ROI (*c.f.*, Section 3.1) for a given pair of images I_1 and I_2.

$$corr = \frac{\sum_{i,j}(R_{I_1}[i,j] - \overline{R}_{I_1}) \cdot (R_{I_2}[i,j] - \overline{R}_{I_2})}{\sqrt{\sum_{i,j}(R_{I_1}[i,j] - \overline{R}_{I_1})^2} \cdot \sqrt{\sum_{i,j}(R_{I_2}[i,j] - \overline{R}_{I_2})^2}}. \quad (6)$$

We considered the color channels R, G, and B in this task because experiments showed that using Y did not provide significant improvement for device linking. Thus, we end up with 27 features in total.

3.2. Open Set Classification

Open set recognition is more than just setting a decision threshold [1, 2, 3]. Our approach begins by learning a classifier from the training set consisting of the positive samples and the available negative samples. Given training data $(\mathbf{x}_i, y_i)$ for $i = 1 \ldots N$, with $\mathbf{x}_i \in \Re^d$ and $y_i \in \{-1, 1\}$, a classifier f is learned such that

$$f(\mathbf{x}_i) = \begin{cases} \geq 0, & y_i = +1 \\ < 0, & y_i = -1. \end{cases} \quad (7)$$

Let X be a training data matrix in which the i^{th} row of X denotes the row vector $\mathbf{x}_i$. Consider that the positive training class consists of feature vectors $\mathcal{P} = \{\mathbf{x}_1^+, \mathbf{x}_2^+, \ldots \mathbf{x}_{n^+}^+\}$ and the negative class (which in itself can contain multiple negative classes) consists of $\mathcal{N} = \{\mathbf{x}_1^-, \mathbf{x}_2^-, \ldots \mathbf{x}_{n^-}^-\}$ where $N = n^+ + n^-$ is the total number of training examples.

We can find a maximum margin separation hyperplane $\vec{w} \cdot \mathbf{x} + b = 0$ (linear case) or $\vec{w} \cdot \phi(\mathbf{x}) + b = 0$ (non-linear case) by means of the classical support

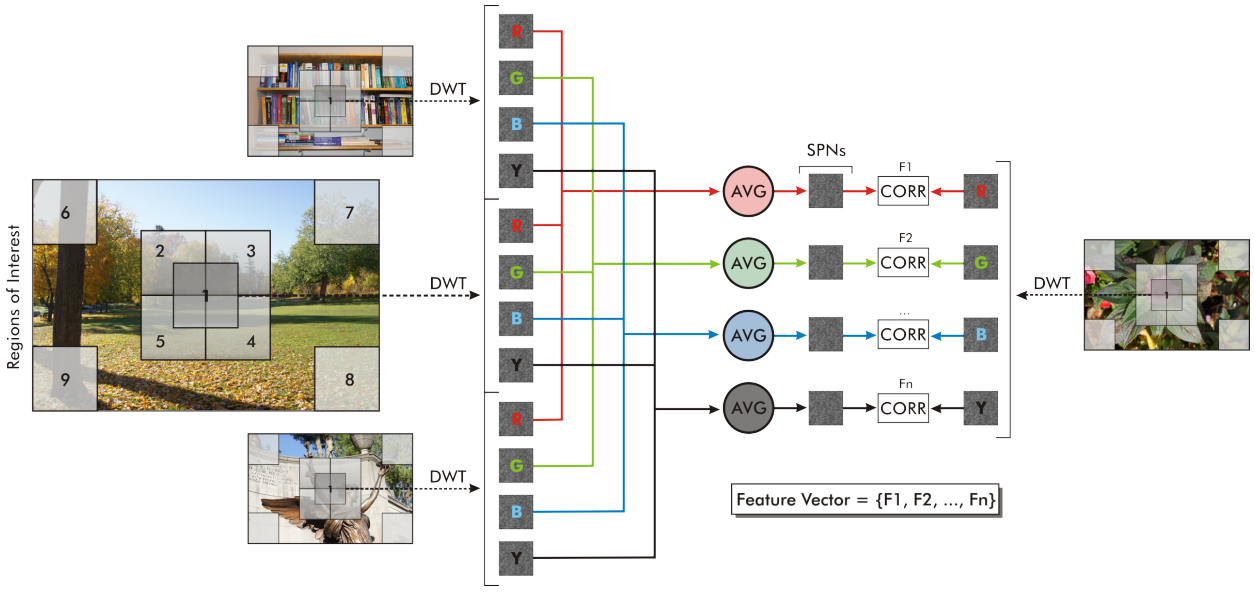

Figure 3: Feature generation takes advantage of multiple image regions, giving us a rich pool of data to consider when learning classifiers that perform well in open set scenarios. For source camera attribution (shown above), we calculate the residual noise for each ROI considering the R, G, B and Y color channels. Next, we generate the feature vector with respect to the correlation between the noise in each ROI and the corresponding noise pattern for each camera. Performing this for all ROIs for one camera, we have 36 features for each image. Device linking proceeds much the same way, except that pairs of images are considered, and we only analyze the R, G, and B color channels, giving us 27 features for each image.

vector machine classification algorithm [33, 25]. The objective of SVM is to find a classifier that is able to separate the data points from $\mathcal{P}$ and $\mathcal{N}$, where $\vec{w}$ is the normal to the hyperplane, b is the bias of the hyperplane such that $|b|/\|\vec{w}\|$ is the perpendicular distance from the origin to the hyperplane, and ϕ is a mapping function from original feature space to a higher dimensional space by means of the kernel trick [33]. After finding a maximum margin separation hyperplane (the solution f) from the training data points X, we have a situation where we have one class of interest as the positive data (*e.g.*, consisting of data points from a suspicious camera) and one or more classes as the negative data (*e.g.*, consisting of data points from other known cameras). According to this model, each data point $\mathbf{x}_i$ during training is at a distance d_i to the decision boundary given the SVM solution and can be classified as class $+1$ if $\vec{w} \cdot \mathbf{x_i} + b \geq 0$ or as -1, otherwise.

SVM uses structural risk minimization[3] [33], allowing it to minimize the risk of misclassification based only on what it knows from the training data. In the open set case, however, many more classes can appear that are part of the overall negative class (*e.g.*, other cameras in the world), which could adversely affect the operation of the classifier during testing. Therefore, given the sensor attribution or device linking classification problem, our objective in the open set scenario is to minimize the risk of the unknown by maximizing the training classification accuracy while minimizing the false positive rate (false matches of other cameras to the camera of interest). We achieve that by solving the following optimization problem:

$$\min\left(\frac{1}{A(X)}\right), \tag{8}$$

where $A(X)$ is the normalized training accuracy given by

$$A(X) = \frac{1}{2}\left(\frac{\sum_{i=1}^{(n^+)} \theta(\mathbf{x_i^+})}{n^+} + \frac{\sum_{j=1}^{(n^-)} \omega(\mathbf{x_j^-})}{n^-}\right) \tag{9}$$

such that

$$\theta(\mathbf{x_i^+}) = \begin{cases} 1, & \text{if } f(\mathbf{x_i^+}) \geq \varepsilon \\ 0, & \text{otherwise;} \end{cases} \tag{10}$$

$$\omega(\mathbf{x_j^-}) = \begin{cases} 1, & \text{if } f(\mathbf{x_j^-}) < \varepsilon \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

[3] Structural risk minimization refers to the inductive principle for model selection used for learning from finite training data sets and solving the problem of finding the maximum margin separation hyperplane.

In summary, Eq. 9 means that we analyze the classification values of all training samples to find the classification accuracy $A(X)$.

With the calculated hyperplane in the initial training step, which represents the best SVM can do based on what it knows during training time, we move the decision hyperplane by a value ε inwards towards the positive class or outwards in the direction of the negative known class(es) in order to account for unknown classes and to minimize future false positive matches. By changing the hyperplane position, we can be more strict about what we know to be positive examples and therefore classify any other data point that is "too different" as negative, or we can be less strict about what we know with respect to the positive class and accept more distant data points as possible positive ones.

We consider ε to move in the interval given by the most positive example (farthest from the decision hyperplane in the positive direction) and the most negative example (farthest from the decision hyperplane in the negative direction). For simplification, we might constrain the interval, as we do in this paper, to be tighter such as $\in [-1, 1]$, so as not to drastically change the initial hyperplane found by the SVM. The ε value represents a movement on the decision hyperplane $\vec{w} \cdot \mathbf{x} + b + \varepsilon = 0$ (the linear case) or $\vec{w} \cdot \phi(\mathbf{x}) + b + \varepsilon = 0$ (the non-linear case considered herein). We loosely call this process Decision Boundary Carving (DBC). An exhaustive search to minimize the training data error defines the value of ε, which we accomplish by solving Eq. 8. Given any data point $\mathbf{z}$ for testing, it is classified as a positive example if $f(\mathbf{z}) \geq \varepsilon$. Fig. 4 depicts the DBC process and Appendix A outlines a high-level algorithm for DBC.

3.3. Open Set Camera Attribution and Device Linking

We can train a classifier for the open set camera attribution problem in a straightforward manner from the methodology just defined. The steps are: (1) collecting feature vectors according to Sec. 3.1 from the positive class of interest and from the known negative classes; (2) training a classifier; and (3) finally performing decision boundary carving for finding the most suitable position for the decision hyperplane according to Section 3.2. For solving the device linking problem while accounting for its open set nature, all we need to do is adapt the feature extraction procedure as explained in Section 3.1. Similar to the camera attribution problem, the final step consists of training a two-class SVM classifier and moving the decision hyperplane via DBC so as to avoid/diminish false matches.

4. Experiments and Results

In this section, we present the experiments performed to validate the proposed method as well as to compare it with state-of-the-art solutions considering the image source attribution and device linking problems. We divide the experiments into two partitions according to the problems of interest.

4.1. Data sets and experimental protocol

We consider a master data pool of 13,210 unique images coming from 400 different digital cameras, with a large portion of the images consisting of unconstrained photos we downloaded from the web. As camera attribution and device linking are conceptually different, we organized this set of images into two datasets[4]:

- **Dataset A** consists of 13,210 images from 400 different cameras. We have physical access to 25 of the cameras and the other 375 cameras are represented by images collected in the wild (Flickr) to simulate what happens in a real world investigation when seizing devices and images. We further organize Dataset A into datasets A_{25} and A_F. A_{25} includes images whose source is within the set of 25 cameras we have physical access to. A_F includes images from Flickr and represents a large number of source cameras. Table 1 provides a comprehensive listing of camera details for the A_{25} dataset[5]. A_{25} contains 4,411 images with an average of 150 images per camera, while A_F contains 8,799 images in total with a varied number of images per camera (reflecting the unconstrained acquisition).

  In the camera attribution experiments, we analyze a scenario in which a "search, seize, and capture" has occurred and a set of cameras and suspicious images were apprehended. The question is then to check whether any of the images come from any of the cameras or from another device not in the apprehended set of cameras. This is an open

[4] All the feature vectors for both datasets as well as the list of used images will be freely available upon acceptance at `http://www.ic.unicamp.br/~rocha/pub/communications`

[5] The list of all cameras from Flickr is included as supplementary material.

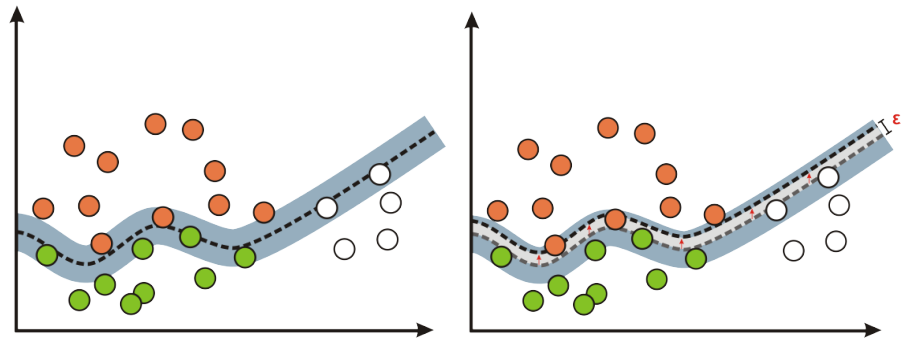

Figure 4: Source camera attribution and device linking using Decision Boundary Carving (DBC). (a) Calculated separation hyperplane, considering the orange and green data points as the known positive and known negative classes, respectively, and the white data points as the unknown classes. The bluish region represents the distance between the margins of the positive and negative support vectors. (b) DBC over the calculated hyperplane, represented by the gray region. The process of carving the decision boundary seeks to minimize the risk of the unknown by minimizing the data error, which is implemented as the normalized accuracy during training $\left(\frac{1}{A(X)}\right)$.

Table 1: Cameras used for all experiments. Images or matched pairs / non-matched pairs of images from cameras 1–15 are the only ones that can be used for training in both the camera attribution and device linking problems. Images or matched pairs / non-matched pairs of images from cameras 16-25 and from cameras in Flickr are always used for testing, and represent the "open set".

	Camera	Native Resolution		Camera	Native Resolution
1	Canon PowerShot SX1-LS	3840 × 2160	**14**	Nikon D40	3008 × 2000
2	Kodak EasyShare c743	3072 × 2304	**15**	Olympus SP570UZ	3968 × 2976
3	Sony Cybershot DSC-H55	4320 × 3240	**16**	Panasonic Lumix DMC-FZ35	4000 × 3000
4	Sony Cybershot DSC-S730	2592 × 1944	**17**	Sony Alpha DSLRA 500L	4272 × 2848
5	Sony Cybershot DSC-W50	2816 × 2112	**18**	Olympus Camedia D395	2048 × 1536
6	Sony Cybershot DSC-W125	3072 × 2304	**19**	Sony Cybershot DSC-W120	3072 × 2304
7	Samsung Omnia	2560 × 1920	**20**	Nikon Coolpix S8100	4000 × 3000
8	Apple iPhone 4 (1)	2592 × 1936	**21**	Sony Cybershot DSC-W330	4320 × 3240
9	Kodak EasyShare M340	3664 × 2748	**22**	Apple iPhone 4 (2)	2592 × 1936
10	Sony Cybershot DSC-H20	3648 × 2736	**23**	Canon Powershot A520	1600 × 1200
11	HP PhotoSmart R727	2048 × 2144	**24**	Apple iPhone 3	1600 × 1200
12	Canon EOS 50d	4752 × 3168	**25**	Samsung Star	2048 × 1536
13	Kodak EasyShare Z981	4288 × 3216	**Flickr**	375 different cameras (brands and models)	Various Resolutions

set attribution problem because the apprehended images may or may not have come from one of the apprehended cameras. However, any solution dealing with such problem needs to be trained using only the apprehended cameras.

For that, we analyze the open set image source attribution problem considering access to sets of 15, 10, 5 and 2 suspect cameras from the A_{25} set. However, we emphasize that in testing, the images can be generated by any of the cameras represented in A_{25} and A_F. For a better analysis of the training variation, we divide the images from cameras 1-15 from A_{25} into five groups and perform a 5-fold cross validation during training. The cross-validation for set A_{25} is intended to analyze how the classifier learning and the decision boundary calculation is affected by variation in the training sets. In each round of the cross-validation, we train the algorithms with four folds and use the fifth one for testing. In addition, the testing set is always complemented by images from cameras 16–25 in A_{25} as well as by images from all cameras in A_F. During training, we never have access to cameras 16–25, nor the cameras that generated the images in A_F.

- **Dataset L** consists of 25,000 pairs of images from two cameras with both in A_{25}, or one camera in A_{25} and another in A_F. We consider 5,000 matched

pairs and 5,000 non-matched pairs from cameras 1-15 in A_{25} (Set L_1), 5,000 matched-pairs and 5,000 non-matched pairs from cameras 16-25 in A_{25} (Set L_2), and 5,000 non-matched pairs from one camera in the set of cameras 16-25 from A_{25} and another camera in A_F (Set L_3). The algorithms are always trained with half of the matched pairs and non-matched pairs in L_1 and tested with the other half of L_1 and all pairs of images in L_2 and L_3. The process is repeated once, switching the training half. The L_2 and L_3 testing represent the open set validation partition, since the matched-pairs and non-matched pairs of images therein come from cameras never used during the training phase of the algorithms.

For SVM classification in all experiments, we use the LibSVM library [34]. We consider only the non-linear case herein with a radial basis function kernel (RBF):

$$RBF(\mathbf{x_i^+}, \mathbf{x_j^-}) = \exp\left(-\gamma \cdot |\mathbf{x_i^+} - \mathbf{x_j^-}|^2\right) \quad (12)$$

We find the best separation hyperplane during training through a grid search by changing the RBF kernel parameter γ (that represents the boundary smoothness between positive and negative samples) and the cost of misclassification, considering just the known samples (positive and known negative classes at training).

4.2. Image source attribution

To measure the effectiveness of the image source attribution analysis, we calculate the accuracy (in %) for each camera considering the relative classification accuracy Acc_R according to

$$Acc_R = \frac{Acc^+ + Acc^-}{2}, \quad (13)$$

which is the average of the percentage of correct classifications during testing for positive (Acc^+) and negative (Acc^-) images for a given camera. The average accuracy Acc_M for each camera is calculated as

$$Acc_M = \frac{1}{z}\sum_{i=1}^{z} Acc_R^i, \quad (14)$$

where $z = 5$ divisions of the 5-fold cross-validation protocol used for training with the A_{25} set. The results we report correspond to the final accuracy Acc_F, calculated as the average over all cameras

$$Acc_F = \frac{1}{N_C}\sum_{i=1}^{N_C} Acc_M^i, \quad (15)$$

where N_C is the number of available cameras during training.

When we consider access to cameras 1–15, it means we train with cameras 1–15 as suspect cameras but the images under investigation can come from any of the 25 cameras in A_{25} as well as from any of the 375 cameras in A_F. For the case with 10 known cameras during training, we performed two experiments (access to cameras 1–10 and separately access to cameras 6–15). For the experiments with five cameras, we considered three different combinations of five cameras (1–5, 6–10, 11–15). For the experiments with two available cameras, we considered seven different combinations (1–2, 3–4, and so forth).

We validate the proposed method in four ways. We refer to our approach considering only ROI #1 (see Fig. 3) as T_1, with ROI #1 plus the open set decision boundary carving (DBC) solution as T_2, our approach considering all ROIs without DBC as T_3, and the complete solution with all regions plus DBC as T_4. For each case, the result is the average of the results for tests considering each combination of cameras. Table 2 shows the comparison of the proposed methods to Lukáš et al.'s [1] and Li's [2] approaches in an open set scenario.

Table 2: Results ($Acc_F \pm \sigma$, in percentage) for 15, 10, 5, and 2 available cameras during training. An open set with 15 out of 400 cameras consists of training on 15 cameras but testing on images from any of the cameras already seen in training and also from either any of the cameras in the set 16-25 never used in training or any camera in A_F (the wild). The number of testing examples change for each scenario. For instance, for each positive testing camera (cameras 1-15 in A_{25}), there are 1×30 positives and 14×30 negatives from the known classes, 10×30 negatives from cameras 16-25 in A_{25} and 8,799 negatives from A_F, for a total of 30 positives and 9,519 negatives. Best results are highlighted in bold.

	Open Set Cameras – Results in %			
	15	**10**	**5**	**2**
LUKAS ET AL. [1]	95.08 ± 2.40	94.70 ± 2.46	95.04 ± 2.34	94.46 ± 2.69
LI. [2]	94.62 ± 2.56	94.06 ± 2.67	94.50 ± 2.49	93.23 ± 2.84
TC-SVM – T_1 Only central ROI	90.95 ± 3.14	91.17 ± 2.62	93.65 ± 2.63	94.29 ± 2.81
TC-SVM – T_2 Central ROI + DBC	95.95 ± 1.70	95.35 ± 1.95	95.84 ± 1.63	94.89 ± 2.01
TC-SVM – T_3 All ROIs without DBC	95.75 ± 1.64	95.69 ± 1.83	96.88 ± 1.48	**96.56** ± 1.65
TC-SVM – T_4 All ROIs + DBC	**97.18** ± 1.63	**96.80** ± 1.63	**97.34** ± 1.16	96.49 ± 1.60

Specifically, for the case with 15 known suspect cameras during training and 400 cameras in testing, we

can see that the open set feature characterization (T_3) with no decision boundary carving slightly improves the classification accuracy when compared to the best baseline (Lukáš et al. [1]). T_3 reduces the classification error by 13%[6]. When we add the decision boundary carving to further deal with the open set nature of the problem, the classification error is reduced by 42.7%. These results show that it is possible to reliably identify image sources in an open set scenario even with hundreds of unknown cameras.

The results also indicate that the DBC implementation does not help in the case of only two suspect cameras when considering all ROIs. However, we still see the improvement in results when we consider more ROIs for the identification, which makes the case for the open set feature treatment we devised in Section 3.1. The approach proposed by Li [2] does not statistically improve the classification results of Lukáš et al. [1] (considering the dataset and the open set evaluation scenario used in this work). We used the Wilcoxon Sign Rank [35] at the 99% confidence interval for statistical significance tests. Any statistical difference between the methods devised by Lukáš et al. [1], Li [2] and our approaches (T_3 and T_4) is shown in Table 3.

Table 4 contains a breakdown for the case with known cameras 1–15 for training and 400 for testing (385 unknown). It shows the true positive rate, as well as the true negative rate with results in $X\% \pm \sigma$ (standard deviation), as well as the raw numbers considering the average of a 5-fold cross validation protocol. Observe that the proposed method demonstrates higher performance than the approaches of Lukáš et al. [1] and Li [2]: we reduce the risk of the unknown considerably. This is reflected in the high number of true negatives (and consequently low false positives) with a very low standard deviation and an increase in the true positives.

4.3. Device linking

For device linking, we present the proposed methods without DBC (T_3) and with DBC (T_4) and compare them to the state-of-the-art method proposed by Goljan et al. [3] (*Goljan Baseline*). We also compared our methods to an extension of Goljan et al. [3] (*Goljan ML Extended*) in order to place it on a common machine learning basis with our approach. We carried out the experiments considering the original and the enhanced residual noise [2] of the images.

Table 4: Breakdown for the Open Set setup with cameras 1–15 from A_{25} for training and 400 cameras for testing from A_{25} and A_F.

	Lukáš et al. [1]	**Li [2]**	**TC-SVM + DBC – T_4**
TP	92.59% ± 5.15 (27.77 / 30)	91.62% ± 5.70 (27.48 / 30)	**95.81%** ± 3.46 (28.74 / 30)
TN	97.56% ± 1.77 (9,277 / 9,519)	97.62% ± 1.55 (9,283 / 9,519)	**98.54**% ± 0.53 (9,370 / 9,519)

The baseline proposed by [3] essentially fails in this open set validation scenario. Regarding this method, a region of size 512 × 512 pixels around the center of each image was considered following the original paper. We carried out two experiments in this case: in the first experiment, we examined how effective the threshold found by the authors was for the dataset used herein. In the second experiment, we calculated the best threshold considering the PSR values for our training set (images from the L_1 set in dataset L), and assessed that value using our test set (containing images from L_1, L_2 and L_3). We found accuracies of only about 50%. A possible reason for such results is that the authors perform zero-padding aiming at matching the resolution of the images, and we just considered the central region of the images. As the images have different native resolutions in our dataset we could not strictly reproduce their validation scenario. Therefore, we extended their method to automatically find the threshold using the same classifier we use in this article: SVM. For that, the classifier is given a 3-d feature vector composed of the result of the correlation values between the pair of images for each color channel as input. The performance improvement is remarkable: going from the original chance baseline to 75.6% (see Goljan ML Extended method in Table 5).

Nonetheless, the proposed methods T_3 and T_4 yield an average accuracy of 87.4% for correctly classifying a random image pair as generated or not by the same camera. This is an improvement of 11.7% and represents a reduction of 52% in the classification error. A Wilcoxon Sign Rank test [35] at the 99% confidence interval shows that the Goljan ML Extended method is statistically better than the baseline, but both methods are statistically worse than T_3 and T_4. In addition, we can see that T_3 and T_4 are very similar, which means that the multi-region feature characterization was enough for open set device linking – the DBC procedure

[6]The error reduction is calculated by dividing the error of the proposed method by the error of the baseline and taking the complement. For instance, suppose a proposed method A has an error of 10% and the baseline has an error of 50%. In this case, we can say A reduced the classification error compared to B by: $1 - (0.1/0.5) = 80\%$.

Table 3: Significance tests between our solutions and the ones proposed by Lukáš et al. [1] and Li. [2]. A '•' means statistically significant differences.

	Lukáš et al. [1]				**Li [2]**				**TC-SVM -** T_3				**TC-SVM + DBC -** T_4			
Approach \ Number of Cameras	**15**	**10**	**5**	**2**	**15**	**10**	**5**	**2**	**15**	**10**	**5**	**2**	**15**	**10**	**5**	**2**
Lukáš et al. [1]	–	–	–	–			•	•	•	•	•	•	•	•	•	•
Li [2]			•	•	–	–	–	–	•	•	•	•	•	•	•	•
TC-SVM - T_3	•	•	•	•	•	•	•	•	–	–	–	–	•	•	•	•
TC-SVM + DBC - T_4	•	•	•	•	•	•	•	•	•	•	•	•	–	–	–	–

Table 5: Summary results for device linking obtained with a 2-fold cross-validation protocol.

Exp. ID	**Info. Used**	**Acc.**	**Std.Dev.**
Goljan Baseline [3]	Original PRNU	51%	2.63
Goljan Baseline [3]	Enhanced PRNU	51.1%	2.14
Goljan ML Extended	Original PRNU	76.3	0.98
Goljan ML Extended	Enhanced PRNU	75.6%	0.97
Proposed T_3	Original PRNU	86.7	0.81
Proposed T_3	Enhanced PRNU	**87.4**%	1.62
Proposed T_4	Original PRNU	86.5	1.31
Proposed T_4	Enhanced PRNU	**87.4**%	2.51

does not play a major role here. Table 5 summarizes the results of all methods.

5. Discussion

Understanding the image source attribution problem in an open set context is an important step towards solving a real world problem. It is not difficult to imagine a situation where an investigator needs to answer the question of whether an apprehended photograph belongs to one out of a possible set of known cameras or to some other unknown camera. Just as important as open set camera attribution, the open set device linking problem presents the investigator with the task of determining if two images were generated by the same camera. Our results for both problems were encouraging, with remarkable reductions in error rates despite the presence of a large open set of unconstrained images from the wild. However, the method we propose is just a first contribution towards a comprehensive solution to this very challenging and poorly understood general problem of open set classification.

As with almost any work, ours can certainly be improved upon. We can extend this research to help combat counter-forensic approaches, as discussed by Goljan et al. [36] and Gloe et al. [22]. The application of the core idea of the proposed feature characterization technique, which is tailored to expanding the amount of information available to define a projection space for a unique class of interest, as well as the principles of the decision boundary carving technique, to other pattern recognition and computer vision problems looks promising.

Finally, it is important to bear in mind that the solutions presented herein are applicable to myriad pattern recognition and vision problems operating in the open set mode. According to Duin and Pekalska [37], the way we approach a problem that contains classes that are ill-sampled, not sampled at all, or are undefined is definitely an open issue. Therefore, we envision our open set solutions will represent possible starting points for approaches to other problems such as face recognition and verification, object recognition and image categorization – all longstanding areas of prime importance in pattern recognition and computer vision.

Acknowledgments

We acknowledge the financial support of FAPESP (Grant #2010/05647-4) CNPq (Grant #304352/2012-8), Microsoft and Samsung. Part of the results presented in this paper were obtained through the project "Unicamp/Samsung", sponsored by Samsung Electronics da Amazônia Ltda., in the framework of law No. 8,248/91.

References

[1] J. Lukáš, J. Fridrich, M. Goljan, Digital camera identification from sensor pattern noise, IEEE Transactions on Information Forensics and Security (TIFS) 1 (2) (2006) 205–214.

[2] C.-T. Li, Source camera identification using enhanced sensor pattern noise, IEEE Transactions on Information Forensics and Security (TIFS) 5 (2) (2010) 280–287.

[3] M. Goljan, J. Fridrich, Identifying common source digital camera from image pairs, in: IEEE Intl. Conference on Image Processing (ICIP), 2007, pp. 14–19.

[4] A. Rocha, W. Scheirer, T. E. Boult, S. Goldenstein, Vision of the unseen: Current trends and challenges in digital image and video forensics, ACM Computing Surveys (CSUR) 42 (26) (2011) 26:1–26:42.

[5] D. Li, Ballistics projectile image analysis for firearm identification, IEEE Transactions on Image Processing (TIP) 15 (10) (2002) 2857–2865.
[6] A. Swaminathan, M. Wu, K. Liu, Component forensics - theory, methodologies and applications, IEEE Signal Processing Magazine 26 (2) (2009) 38–48.
[7] K. Kurosawa, K. Kuroki, N. Saitoh, CCD fingerprint method – identification of a video camera from videotaped images, in: IEEE Intl. Conference on Image Processing (ICIP), 1999, pp. 537–540.
[8] A. E. Dirik, H. T. Sencar, N. Memon, Digital single lens reflex camera identification from traces of sensor dust, IEEE Transactions on Information Forensics and Security (TIFS) 3 (3) (2008) 539–552.
[9] P. Chiang, N.Khana, A. K. Mikkilineni, M. V. O. Segovia, S. Suh, J. P. Allebach, G. T. C. Chiu, E. J. Delp, Printer and scanner forensics, IEEE Signal Processing Magazine 72 (2) (2009) 72–83.
[10] E. Kee, H. Farid, Printer profiling for forensics and ballistics, in: ACM Workshop on Multimedia and Security, Vol. 10, 2008, pp. 3–10.
[11] N. Khanna, A. K. Mikkilineni, G. T. C. Chiu, J. P. Allebach, E. J. Delp, Scanner identification using sensor pattern noise, SPIE Security, Steganography and Watermarking of Multimedia Contents (SSWMC) 6505.
[12] N. Khanna, A. K. Mikkilineni, E. J. Delp, Scanner identification using feature-based processing and analysis, IEEE Transactions on Information Forensics and Security (TIFS) 4 (1) (2009) 123–139.
[13] A. Popescu, H. Farid, Exposing digital forgeries in color filter array interpolated images, IEEE Transactions on Signal Processing (TSP) 53 (10) (2005) 3948–3959.
[14] M. Kharrazi, H. Sencar, N. Memon, Blind source camera identification, in: IEEE Intl. Conference on Image Processing (ICIP), Singapore, 2004, pp. 709 – 712.
[15] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, N. Saitoh, Methods for identification of images acquired with digital cameras, Enabling Technologies for Law Enforcement and Security 4232 (2001) 505–512.
[16] F. de Oliveira Costa, M. Eckmann, W. J. Scheirer, A. Rocha, Open set source camera attribution, in: SIBGRAPI Conference on Graphics, Patterns and Images, 2012, pp. 1–8.
[17] M. Goljan, J. Fridrich, T. Filler, Large scale test of sensor fingerprint camera identification, in: Proc. SPIE, Vol. 7254, 2009, pp. 72540I–72540I–12.
[18] M. Chen, J. Fridrich, M. Goljan, J. Lukáš, Determining image origin and integrity using sensor noise, IEEE Transactions on Information Forensics and Security (TIFS) 3 (1) (2008) 74–90.
[19] C.-T. Li, Unsupervised classification of digital images using enhanced sensor pattern noise, in: IEEE Intl. Symposium on Circuits and Systems (ISCAS), 2010, pp. 3429–3432.
[20] R. Caldelli, I. Amerini, F. Picchioni, M. Innocenti, Fast image clustering of unknown source images, in: IEEE Intl. Workshop on Information Forensics and Security (WIFS), 2010, pp. 1–5.
[21] Y. Sutcu, S. Bayran, H. Sencar, N. Memon, Improvements on sensor noise based source camera identification, in: IEEE Intl. Conference on Multimedia and Expo, 2007, pp. 24–27.
[22] T. Gloe, M. Kirchner, A. Winkler, R. Böhme, Can we trust digital image forensics?, in: ACM Multimedia, 2007, pp. 78–86.
[23] R. Caldelli, I. Amerini, A. Novi, An analysis on attacker actions in fingerprint–copy attack in source camera identification, in: IEEE Intl. Workshop on Information Forensics and Security (WIFS), Foz do Iguaçu, 2011, pp. 1–6.
[24] P. Phillips, P. Grother, R. Micheals, Handbook of Face Recognition, Springer, 2005, Ch. Evaluation Methods on Face Recognition, pp. 329–348.
[25] C. Cortes, V. Vapnik, Machine Learning, 20th Edition, Kluwer, 1995, Ch. Support-Vector Networks, pp. 273–297.
[26] W. Scheirer, A. Rocha, A. Sapkota, T. E. Boult, Towards open set recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) 36 (7).
[27] B. Wang, X. Kong, X. You, Source camera identification using support vector machines, in: Advances in Digital Forensics V, Vol. 306 of IFIP Advances in Information and Communication Technology, Springer Boston, 2009, pp. 107–118.
[28] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson, Estimating the support of a high-dimensional distribution, Neural Computation 13 (7) (2001) 1443–1471.
[29] X. Zhou, T. Huang, Relevance feedback in image retrieval: A comprehensive review, Multimedia Systems 8 (6) (2003) 536–544.
[30] C.-T. Li, R. Satta, On the location-dependent quality of the sensor pattern noise and its implication in multimedia forensics, in: Proc. IV Intl. Conference on Imaging for Crime Detection and Prevention (ICDP), 2011, pp. 1–6.
[31] D. B. Goldman, J. H. Chen, Vignette and exposure calibration and compensation, in: Intl. Conference on Computer Vision (ICCV), 2005, pp. 899–906.
[32] X. Wang, Z. Weng, Scene abrupt change detection, in: Canadian Conference on Electrical and Computing Engineering, 2000, pp. 880–883.
[33] C. M. Bishop, Pattern Recognition and Machine Learning, 1st edition, Springer, 2006.
[34] C. Chang, C. Lin, LIBSVM: A library for support vector machines, Transactions on Intelligent Systems and Technology (TIST) 2 (3) (2011) 27:1–27:27.
[35] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics Bulletin 1 (6) (1999) 80–83.
[36] M. Goljan, J. Fridrich, M. Chen, Defending against fingerprint-copy attack in sensor-based camera identification, in: IEEE Transactions on Information Forensics and Security (TIFS), 2011, pp. 227–236.
[37] R. P. W. Duin, E. Pekalska, Open issues in pattern recognition, in: Intl. Conference on Computer Recognition Systems (CORE), 2005, pp. 27–42.

Appendix A. Decision Boundary Carving Algorithm

Algorithm 1 shows the pseudo-code for the decision boundary carving process described in this paper. The refinement process in which we further separate the training data into the actual training set (for finding the SVM parameters $\vec{w}$ and b) and validation set (for optimizing the open set threshold ε) is not shown for the purpose of simplicity.

Algorithm 1 Decision Boundary Carving (DBC)

Input: $\mathcal{P}, \mathcal{N}$ ▷ Set of elements of the positive class of interest and the known negative class(es), respectively
Output: Decision hyperplane parameters $\vec{w}$ and b, and the open set threshold ε

$(\vec{w}, b) \leftarrow$ SVM-Training($\mathcal{P}$, $\mathcal{N}$); ▷ Calculating the SVM hyperplane parameters
$\mathbf{C} \leftarrow$ Classification($\vec{w}$,b, $\mathcal{P}$, $\mathcal{N}$); ▷ Obtaining the decision scores
min $\leftarrow$ lowest-decision-score($\mathbf{C}$);
max $\leftarrow$ highest-decision-score($\mathbf{C}$);
$\mathcal{D} \leftarrow +\infty$ ▷ Setting the initial data error to a maximum value
For $\varepsilon' \leftarrow$ min **to** max **do** ▷ ε' spans possible scores in $\mathbf{C}$ (increments of 10^{-4} herein)
  $(\mathrm{A}^+, \mathrm{A}^-) \leftarrow 0$
  For all $\mathbf{x}^+ \in \mathcal{P}$ **do**
    $\mathrm{A}^+ \leftarrow \mathrm{A}^+ + \theta(\mathbf{x}^+, \varepsilon')$; ▷ True positives for this particular position of the hyperplane (Eq. 10)
  End For
  For All $\mathbf{x}^- \in \mathcal{K}$ **do**
    $\mathrm{A}^- \leftarrow \mathrm{A}^- + \omega(\mathbf{x}^-, \varepsilon')$; ▷ True negatives for this particular position of the hyperplane (Eq. 11)
  End For
  $A_X \leftarrow \frac{1}{2}\left(\frac{A^+}{|\mathcal{P}|}, \frac{A^-}{|\mathcal{N}|}\right)$; ▷ Normalized averaged accuracy (Eq. 9)
  $\mathcal{D}' \leftarrow \frac{1}{A_X}$;
  If $\mathcal{D}' < \mathcal{D}$ **then**
    $\mathcal{D} \leftarrow \mathcal{D}'$;
    $\varepsilon \leftarrow \varepsilon'$;
  End If
End For
Return $(\vec{w}, b, \varepsilon)$