An In-Depth Study on Open-Set Camera Model Identification

PEDRO RIBEIRO MENDES JÚNIOR [1,2], **LUCA BONDI** [2], **(Student Member, IEEE),**
PAOLO BESTAGINI [2], **(Member, IEEE), STEFANO TUBARO** [2], **(Senior Member, IEEE),**
AND ANDERSON ROCHA [1], **(Senior Member, IEEE)**
[1]Institute of Computing, University of Campinas, Campinas 13083-852, Brazil
[2]Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy

Corresponding author: Pedro Ribeiro Mendes Júnior (pedrormjunior@gmail.com)

ABSTRACT Camera model identification refers to the problem of linking a picture to the camera model used to shoot it. In this paper, as this might be an enabling factor in different forensic applications to single out possible suspects (e.g., detecting the author of child abuse or terrorist propaganda material), many accurate camera model attribution methods have been developed. One of their main drawbacks, however, are a typical closed-set assumption of the problem. This means that an investigated photograph is always assigned to one camera model within a set of known ones present during the investigation, i.e., training time. The fact that a picture can come from a completely unrelated camera model during actual testing is usually ignored. Under realistic conditions, it is not possible to assume that every picture under analysis belongs to one of the available camera models. In this paper, to deal with this issue, we present an in-depth study on the possibility of solving the camera model identification problem in open-set scenarios. Given a photograph, we aim at detecting whether it comes from one of the known camera models of interest or from an unknown one. We compare different feature extraction algorithms and classifiers especially targeting open-set recognition. We also evaluate possible open-set training protocols that can be applied along with any open-set classifier, observing that a simple alternative among the selected ones obtains the best results. Thorough testing on independent datasets show that it is possible to leverage a recently proposed convolutional neural network as feature extractor paired with a properly trained open-set classifier aiming at solving the open-set camera model attribution problem even on small-scale image patches, improving over the state-of-the-art available solutions.

INDEX TERMS Camera model identification, image forensics, open-set recognition, open-set training protocol.

I. INTRODUCTION

From social networks to media sharing platforms, digital pictures are spreading all over the Internet at an overgrowing pace. However, a major drawback of this phenomenon is the diffusion of illicit or illegal material online, specially visual content. In order to fight this trend, multimedia forensic researchers have focused on the development of numerous solutions aiming at inferring pieces of information related to the acquisition and editing history of images [1]–[3], among others.

A common problem of interest for forensic analysts is camera model identification. This means being capable of detecting which camera model has been used to shoot a given digital photograph based solely on its content. Indeed, this is a first step toward tracking down the author of distributed illicit contents [4] (e.g., pictures related to acts of violence, images linked to terrorist behavior, sexually exploitative imagery of children, among others). Given the social relevance of this problem, in the last few years, a continuous effort has been put forward to the development of more accurate and

---

The associate editor coordinating the review of this manuscript and approving it for publication was Irene Amerini.

efficient camera model identification solutions. These can be broadly split into two categories: (i) model-based methods leveraging the study of characteristic traces left behind by specific operations applied by different camera models on acquired images; and (ii) data-driven methods based on machine-learning techniques that seek to "learn" the patterns of such telltales automatically. Considering the first category, we can cite methods relying on traces left by color filter array (CFA) interpolation [5]–[7], on histogram equalization footprints [8], on traces left by camera lenses [9], and on characteristic noise analysis [10]. Considering the second category, in turn, we can cite the works of Chen and Stamm [11], Marra *et al.* [12], and Tuama *et al.* [13], which extract statistical features in the pixel-domain to train supervised machine-learning classifiers specialized at the problem. More recently, relying upon advancements on deep learning techniques, data-driven solutions based on Convolutional Neural Networks (CNNs) have outperformed prior art [14]–[18], and are becoming an area's staple.

The drawback of all aforementioned data-driven techniques—or, more precisely, the evaluation setup to validate such techniques—is that they mainly cope with camera model identification in a closed-set setup. This means that a finite set of camera models is considered when designing the solution, and each image is attributed to one of these models. However, oftentimes analysts must work in open-set scenarios. This means that the investigator must also be able to recognize whether an image does not belong to any of the known models of interest [4].

In this vein, we present herein an in-depth study on open-set camera model attribution based on a supervised learning pipeline. Specifically, we focus on methodologies that perform an analysis at patch level rather than on the whole image, as this opens the door to future development of tampering detection and localization methods as shown by Bondi et al. [19]. To the best of our knowledge, open-set camera model attribution has only been introduced by Gloe [20] and later on approached by Bayar and Stamm [21]. Bayar and Stamm [21] focus on an open-set binary detection problem, i.e., detecting whether an image comes from a known or unknown camera model. Conversely, we aim to solve the joint problem of (i) detecting whether the image under analysis comes from a known or from an unknown camera model and (ii) determining the image source model when it comes from the set of known models.

In previous work [22], a general open-set classifier have been proposed along with *cross-class validation*, which is a method tailored to open-set scenarios that aims at searching for the parameters of the proposed open-set classifier. In parallel, another previous work [23], also proposing an open-set classifier, introduced a *parameter optimization* procedure that is also tailored at searching the parameters of their proposed classifier, which shares the same essence of the cross-class validation. In the latter work, authors have suggested as future work the employment of their parameter optimization method as a general grid-search procedure that could be applied to any

open-set classifier. In our work, we follow this direction and we evaluate what we call Closed training protocol (the traditional form) and the Open training protocol (with the same essence of the cross-class validation [22] and the parameter optimization [23]). We further study those alternatives and we formalize and evaluate what we call the Network Open (NetOpen) training protocol, specifically tailored to situations in which deep features are employed. As we shall see later on along with the presented results, the equivalence of Open and NetOpen indicates the Open training protocol as the best and cheaper alternative in terms of data required to be employed.

In light of these considerations, our key contributions are the following:

- We study the open-set camera model identification problem analyzing state-of-the-art open-set classification methods.
- We evaluate the effectiveness of CNNs features, compared to hand-crafted ones, for per-patch classification in open-set setups.
- We formalize and evaluate open-set training protocols applied to open-set classification methods during training for proper estimate of parameters for the open-set scenario.
- We carry out the first large-scale testing on the open-set camera model identification problem considering independent datasets and several algorithms, also comparing with known solutions in the literature [21].

The best evaluated solution for the problem combines a deep feature extraction method and a state-of-the-art open-set classifier trained with an open-set training protocol of intermediate complexity. This solution works on 64×64 color patches, making it useful for forgery localization techniques [19]. Moreover, it is capable of reaching state-of-the-art accuracy also in the closed-set framework.

The rest of the paper is structured as it follows. Section II formally introduces the camera model identification problem under different points of view. Section III provides all the details about the algorithmic pipeline used in our evaluation. Section IV reports information about the considered experimental setup. Section V presents the performed experiments and achieved results. Finally, Section VI concludes the paper.

II. OPEN-SET CAMERA MODEL IDENTIFICATION PROBLEM

In this section, we introduce the problem of camera model identification, from the closed-set to the open-set one faced in this paper.

Camera model identification generally refers to the problem of assigning an image, in a blind fashion, to the camera model that was used to shoot it. This means that no watermarks or side information such as header or EXIF data are used, assuming they will not be available during investigation. Depending on the considered constraints, camera model identification can be cast into different kinds of problems,
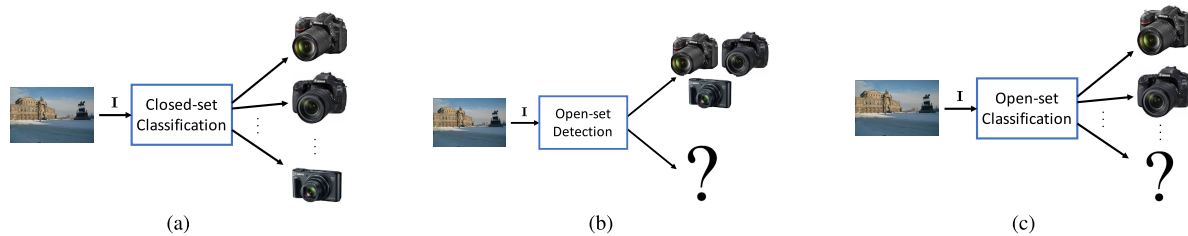
FIGURE 1. Different camera model identification problem formulations. (a) In the closed-set problem, an image is attributed to a known camera model. (b) In the open-set detection problem, an image is detected as belonging either to the known set or the unknown set of models. (c) In the open-set classification problem faced in this work, an image is detected as belonging to the known or unknown model sets and, in the first case, the correct model is also estimated.

as shown in Figure 1. In the following, we report the main differences between these problem formulations.

A. CLOSED-SET CLASSIFICATION

Closed-set camera model classification is the problem of assigning an image to a camera model within a known set of possible models, as depicted in Figure 1(a). In this scenario, it is required to assume that the investigator is sure that the camera model of the picture under analysis belongs to the set of candidate models.

Formally, let **I** be a color image acquired with the camera model identified by label c. Consider further $\mathcal{C}_{\text{known}}$ as the set of labels c belonging to the known camera model dataset, e.g., available to the analyst when developing the solution. The goal in closed-set camera model identification is to estimate the label $\hat{c} \in \mathcal{C}_{\text{known}}$ associated to the picture under analysis.

This is by far the most widely considered scenario in the literature [4]. However, closed-set classification is bound to fail whenever the analyst has no full knowledge on all the possible used camera models: in real-case open-set scenarios, it happens that $c \in \mathcal{C}_{\text{known}} \cup \{c_0\}$, in which c_0, $c_0 \notin \mathcal{C}_{\text{known}}$, is the unknown label that represents any unknown class.

B. OPEN-SET DETECTION

Relaxing the constraint of knowing all possible camera models, we enter the open-set realm. Indeed, in an open-set scenario, the image under analysis can belong to either known or unknown camera models. In particular, we refer to open-set camera model detection as the problem of detecting whether an image belongs to the set of known models, or to the set of unknown ones, as depicted in Figure 1(b).

Formally, the goal of open-set camera model detection is to estimate whether $c \in \mathcal{C}_{\text{known}}$ or $c \notin \mathcal{C}_{\text{known}}$ for a given image **I**. This is basically a two-class classification problem that does not provide the analyst with information on the actual used camera model. To infer the possible used camera model, an open-set detection solution should be paired with a subsequent step of closed-set classification, as proposed by Bayar and Stamm [21].

C. OPEN-SET CLASSIFICATION

The most complete camera model identification problem formulation is that of open-set classification. As a matter of fact, this refers to the problem of jointly estimating whether the image under analysis comes from a camera in the known set of models or from an unknown model and, if condition one holds, also detecting which model it is, as depicted in Figure 1(c).

Formally, the goal of open-set camera model identification is to estimate $\hat{c} \in \mathcal{C}_{\text{known}} \cup \{c_0\}$ for a given image **I**.

Typically, to properly develop an open-set classification solution, three different kinds of data are employed:

- **Known data (train and test):** images shot with models $c \in \mathcal{C}_{\text{known}}$ that the analyst must correctly detect and classify.
- **Known-unknown data (optional; train and test):** images shot with models available at training time but assumed as unknown in order to model unknown camera models at algorithm validation time. Those data might or might not be available.
- **Unknown-unknown data (test only):** images shot with models $c \notin \mathcal{C}_{\text{known}}$ and not used for either training or validation, used to properly evaluate a method's performance in the wild. Those data only appear for classification once the classifier is trained.

Open-set classification is by far the most complete problem formulation of the overall camera model identification problem. In this paper, we present an algorithmic pipeline to solve this problem, deeply analyzing each building block of the algorithm in all combinations of the alternatives.

Previous works in open-set camera model identification have not fully evaluated the multiclass open-set classification problem. Bayar and Stamm [21] have considered the performance of the classification methods for detecting known vs unknown and, independently, the closed-set classification performance among the classes. In this latter evaluation, the classifiers work in a closed-set scenario, i.e., they never predicts as unknown. It is worth considering that the accuracy in a problem as described in Section II-C tends to be smaller than considering, independently, the detection accuracy and the closed-set accuracy of the methods without the option for rejection, as in the open-set classification problem the classification methods can perform the following types of error: misclassification, false unknown, and false known [23].

III. EVALUATION PIPELINE

In this section, we provide all the details about the factors we evaluate in this work. We first provide an overview of the
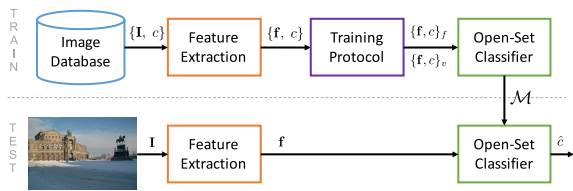
FIGURE 2. Supervised pipeline for open-set camera model identification. At training time (top), a classifier learns how to associate features **f** extracted from labeled images **I** to labels *c*. At testing time (bottom), the learned model $\mathcal{M}$ is used to estimate the label $\hat{c}$ from the image under analysis.

overall algorithmic pipeline. Then, we focus on each separate block of it, reporting information about all methodologies employed in this paper.

A. PIPELINE

To solve open-set camera model attribution, we study the possibility of exploiting a supervised classification strategy leveraging image descriptors tailored to capture camera-based traces proposed in the closed-set scenario literature. Specifically, we follow the pipeline depicted in Figure 2, which is composed by three main modules: (i) a feature extractor, (ii) a training protocol for preparing training data, and (iii) an open-set classifier. For each module, we investigate the possibility of using different strategies.

Feature extraction consists in computing a discriminative feature vector **f** from an image **I**. The feature extractor algorithm is tuned to obtain characteristic camera model information while compacting data dimensionality. Feature vectors **f** extracted from pictures sharing the same camera model should be similar. Conversely, feature vectors **f** extracted from images shot with different models should be, ideally, strongly dissimilar.

Open-set classifiers, as we shall see, tend to associate a bounded region of the feature space to the known classes. A recent work [23] has shown that the split of training data for parameter search can have an influence on the final model obtained by an open-set classifier. The training protocol splits the training data $\{\mathbf{f}, c\}$ into fitting data $\{\mathbf{f}, c\}_f$ and validation data $\{\mathbf{f}, c\}_v$ for parameters search, as depicted in Figure 2. This is a delicate step, as a good open-set classifier must ''learn'' its parameters taking into account the risk of the unknown, not just the empirical risk measured on known data [24]. In essence, prominent alternatives at this stage aim at employing part of the known training data as known-unknown data, as a form of simulation of the unknown.

The role of an open-set classifier is to learn a mapping between feature vectors **f** and camera labels *c*. This mapping is learned at training time by observing several different pairs $\{\mathbf{f}, c\}$ for many different images and *c* values, $c \in \mathcal{C}_{\text{known}}$. The open-set classifier partitions the space spanned by all possible vectors **f**, associating different regions of the feature space to different labels $c \in \mathcal{C}_{\text{known}} \cup \{c_0\}$.

Once the system has been fully trained, it can be deployed. Whenever a new image **I** under investigation is considered,

a feature vector **f** is extracted. The open-set classifier model $\mathcal{M}$ is employed to predict the vector **f** with one class label $\hat{c} \in \mathcal{C}_{\text{known}} \cup \{c_0\}$.

B. FEATURE EXTRACTORS

Different feature extractors for camera-related features have been proposed in the literature. We decided to focus on recently proposed ones that have shown good performance in closed-set camera model attribution setups.

1) RICH FEATURES

Fridrich and Kodovsky [25] have proposed the use of statistical descriptors known as rich features for steganalysis. Rich features are obtained by preprocessing an image through high-pass filtering, quantization and truncation. The rich feature vector is then computed by counting the occurrences of different pixel group combinations. The use of rich features has subsequently proved successful for other forensic applications, from tampering detection [26] to camera model attribution [12]. We denote $\mathbf{f}_{\text{rich}} \in \mathbb{R}^{338}$ as the rich feature vector referred to as SPAM by Marra *et al.* [12] for camera model identification. It has already proved to be more discriminative than those proposed by Gloe [20], Xu and Shi [27], and Celiktutan *et al.* [28] as shown by Marra *et al.* [12].

2) CFA FEATURES

As shown by Chen and Stamm [11], the concept of rich features can be extended to work across different image color planes. Chen and Stamm [11] have shown that it is possible to capture characteristics related to color filter arrays (CFA) for camera model identification. For this reason, we denote $\mathbf{f}_{\text{cfa}} \in \mathbb{R}^{1372}$ as the CFA-based feature vector proposed by Chen and Stamm [11]. As shown by Bondi et al. [15], this can be considered a baseline solution especially when large images are concerned.

3) CNN-DERIVED FEATURES

We adopt as a data-driven method the CNN proposed by Bondi et al. [15] with an architecture comprising four convolutional layers followed by two inner product layers. It has been successfully applied to attribute images to 18 different camera models using 64×64 patches as input. In principle, the output of each CNN layer can be employed as a feature vector **f**. We employ three layers in this work: (i) $\mathbf{f}_{\text{conv}} \in \mathbb{R}^{128}$, obtained after the last convolutional layer; (ii) $\mathbf{f}_{\text{ip1}} \in \mathbb{R}^{128}$, obtained after the first inner product layer; and (iii) $\mathbf{f}_{\text{ip2}} \in \mathbb{R}^{n}$, obtained after the second inner product layer, where $n = |\mathcal{C}_{\text{known}}|$ is the cardinality of the set of known cameras (18 in our experiments, as in the work of Bondi *et al.* [15]).

C. TRAINING PROTOCOLS

To train open-set classifiers, a set of hyper-parameters must be tuned through some method of parameter search to maximize classification accuracy and general-
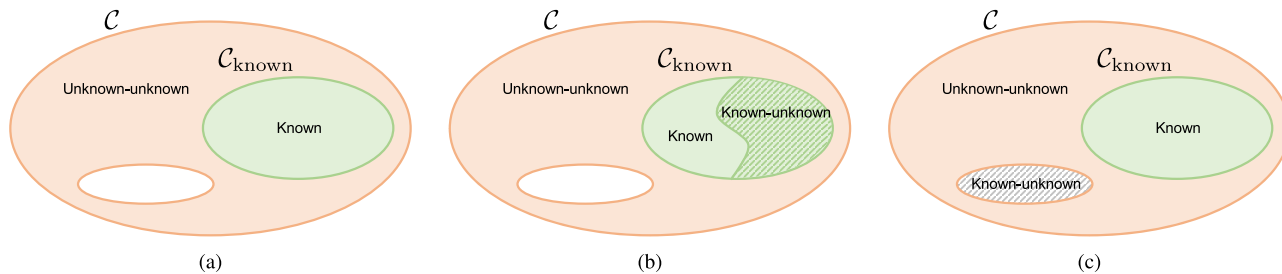
FIGURE 3. Different training strategies: (a) Closed, (b) Open, and (c) NetOpen. The set $\mathcal{C}$ contains all camera models, whereas $\mathcal{C}_{known}$ only contains known models. Green regions represent data for training the CNN employed for feature extraction. Striped regions represent known-unknown camera models employed for classifiers' parameter search and also for training final model in case of (b). Orange regions represent unknown-unknown camera models that appear only for testing. Blank regions are never employed for testing for fair strategies comparison. Although not represented, test set also contains instances from known camera models, for proper evaluation of the classifiers.

ization/specialization capabilities of the employed method. A typical way to do this consists in splitting training data $\{\mathbf{f}, c\}$ into fitting $\{\mathbf{f}, c\}_f$ and validation $\{\mathbf{f}, c\}_v$ data. The selected classifier is then trained on fitting data using different sets of hyper-parameters. Finally, the parameters which model provides the highest accuracy on the set of validation data are selected. The final model is generated on the entire training set with those parameters and results are reported on images belonging to a completely separate (independent) test dataset. In this work, we explore three different training strategies for open-set classifiers. The introduction of this stage in the pipeline was inspired by the work of Mendes Júnior et al. [23], which pointed out their parameter optimization as a general form of grid search for future investigation. In Figure 3, we depict those alternatives as described below.

1) CLOSED STRATEGY

Depicted in Figure 3(a), this is the simplest training strategy, in which no knowledge on the unknown classes is simulated. Indeed, both fitting and validation datasets contain samples from all n known classes (i.e., camera models), and no instance from known-unknown data is used in validation. In other words, parameter search is performed simulating a closed-set setup. This means that the classifier will set the boundaries for each class in the feature space taking into account only the empirical risk aiming at optimizing the separability of the known classes.

2) OPEN STRATEGY

Depicted in Figure 3(b), in order to let the classifier better tune against unknown samples, a straightforward strategy consists in training the classifier on known data, and tuning it considering both the presence of known and known-unknown samples. When the open strategy is selected, $\frac{n}{2}$ of the classes are employed as known and the other $\frac{n}{2}$ are employed as known-unknown in validation. The classifier fitting procedure is carried out on the $\frac{n}{2}$ known classes, however, validation during parameter search is carried out on all n classes, i.e., known and known-unknown camera models. In doing that, parameter search is performed simulating an

open-set setup. After the best parameters are obtained, the final model is trained with all n known training classes to provide a fair comparison with the Closed strategy, i.e., the same number of classes to correctly detect is employed.

3) NETOPEN STRATEGY

Depicted in Figure 3(c), the NetOpen strategy employs unknown data—from the point of view of the network used for feature extraction—as known-unknown data for validation. Dealing with data-driven features (i.e., those extracted using a CNN), special attention must be given to the fact that the CNN, as a feature extractor, must also be trained and validated on the known classes in order to enable discrimination within the set of known camera models.

This strategy considers that the CNN has been separately trained using all available n known classes. The validation set employed during the CNN training process comes from the set of n known classes—as it also happens with Open and Closed strategies. For NetOpen, to better guide the choice of classifiers' parameters, additionally to the n known classes, the validation set also includes samples from extra known-unknown classes, i.e., classes never employed for CNN training or validation. Parameter search of the classifiers is carried out using all known data along with those extra known-unknown data. Finally, when hyper-parameters have been selected, the final model training is performed using just the n known classes, for a paired experiment with the other strategies. In doing that, parameter search is performed simulating an open-set setup also in the point of view of the network.

This approach is appropriate for use with CNN-derived features, however, for the sake of fairness, those extra classes, that are known-unknown from the point of view of the network, are also employed in experiments with $\mathbf{f}_{rich}$ and $\mathbf{f}_{cfa}$ features when NetOpen strategy is applied.

D. OPEN-SET CLASSIFIERS

In the open-set scenario, a classifier should be able to assign one or more bounded regions in the feature space for each known class. In contrast, closed-set classifiers
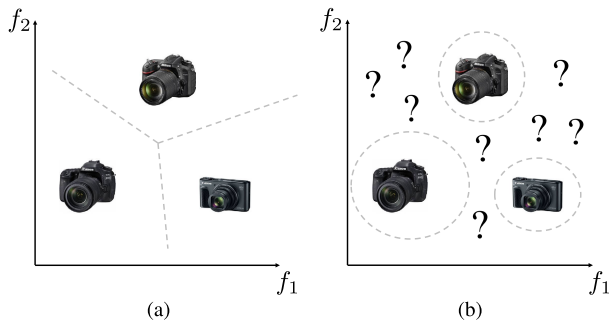
FIGURE 4. Illustration of a three-class classifier in (a) closed- and (b) open-set configurations, considering a two-dimensional feature space $[f_1, f_2]$. The open-set classifier partitions the feature space with bounded regions referring to the known classes in order to enable rejection of unknown classes. In contrast, a closed-set classifier tends to classify any point in the feature space as belonging to one of the known classes.

simply splits unbounded portions of the feature space to each of the known classes. This concept is illustrated in Figure 4.

In this work, we employ for evaluation multiple open-set classifiers available in the literature. Support Vector Machines (SVM) have been applied in the literature to solve various classification problems, including open-set ones in recent works. Traditional SVM can be straightforwardly employed for open-set problems by means the one-vs-all [29] multiclass-from-binary approach [23]: when a feature vector **f** is classified as negative by all binary SVMs that compose the multi-class classifier, then **f** is rejected as unknown. Alternatively, One-class SVM (OCSVM) can also be easily used in open-set setups, as it focuses on carving a decision boundary around known classes, thus points related to unknown classes can be rejected. The same all-negative criterion can be employed for any one-class classifier [30], [31]. Additionally, other methods derived from SVM have been proposed in the literature specifically for open-set problems. In this work, we considered the Weibull-calibrated SVM (WSVM) [32], Decision Boundary Carving (DBC) [33], [34], Specialized Support Vector Machines (SSVM) [35], and SVM with Probability of Inclusion (PISVM) [22].

In addition to these SVM-based approaches, we also consider the Open-Set Nearest Neighbors (OSNN) classifier proposed by Mendes Júnior et al. [23]. This is a recently proposed technique that extends upon the classic nearest neighbors approach. The main rationale behind this method is to avoid relying on raw similarity scores for thresholding. Rejection of unknown instances is accomplished through the of ratio of similarity scores instead. Furthermore, we also consider the classifiers employed by Bayar and Stamm [21], i.e., extremely randomized trees, a.k.a., Extra-Trees (ET) [36], SVM with Platt's probability for rejection (PSVM), thresholding softmax probability (SOFTMAX), and Nearest Class Mean with cosine distance (NCM). Also, by suggestions of previous work [21], we employ a 2-phase SVM (2PSVM) which consists on having a OCSVM for solving the known vs unknown problem, then, if the test instance is classified as known, a PSVM is employed for

choosing the class, otherwise the image is classified to an unknown model.

IV. EXPERIMENTAL SETUP
In this section, we provide details regarding the employed datasets and evaluation metrics.

A. DATASETS
To evaluate all tested methodologies thoroughly, it is important to consider a large enough image database. In this work, we merged three different datasets freely available from previous work.

1) DRESDEN IMAGE DATABASE [37]
This dataset contains almost 17000 images from 27 different camera models. Exactly as in the work of Bondi *et al.* [15], we selected 13000 images from 18 models[1] as the set of images from known camera models. This set was split in training, validation, and test sets [15]. The training set was used to train the CNN-based feature extractor and all classifiers. All images from remaining models—not considered in the subset of 18 models previously selected—have been considered as known-unknown along with the NetOpen strategy and ignored for both Closed and Open strategies.

2) IMAGE SOURCE ATTRIBUTION UNICAMP (ISA UNICAMP)[2]
This dataset contains around 9000 images from 35 camera models. All images from models not overlapping with Dresden Image Database have been selected as unknown-unknown models for the test set in the open-set experiments.

3) FLICKR UNICAMP[2]
This dataset comprises around 11000 images from more than 250 camera models. Differently from previously mentioned datasets, these images have been downloaded from Flickr[3] image hosting service. To avoid dealing with images from the same camera taken at different resolutions, only images at maximum resolution for each model have been selected. All images have been considered as belonging to unknown-unknown camera models in test set for the open-set experiments.

As performed by Bondi *et al.* [15], we obtain, in a content-aware way, 32 non-overlapping 64×64-pixel patches from each image. Provided results are based on majority voting after classification per patch. All patches coming from the same image have been carefully placed only into one of training, validation, and test sets in order to avoid overfitting problems and training/testing contamination.

---

[1]We considered Nikon D70 and Nikon D70s on the same single class due to the negligible differences between them, as reported by Gloe and Böhme [37].

[2]Available at: http://www.recod.ic.unicamp.br/~filipe/dataset [34].

[3]Available at: https://www.flickr.com.

B. METRICS

As evaluation metrics, we employ a set of commonly used ones, as well as others recently proposed for open-set scenario [21], [23]. In particular, we consider different definitions of accuracy and f-measure. Concerning accuracy, we employ the following definitions:

1) ACCURACY ON KNOWN SAMPLES AKS [23]

This is the accuracy in correctly attributing images from known models to the actual models. This metric encompasses two kinds of misclassification errors: known-model images attributed to unknown class (false unknown) and known-model images attributed to wrong known classes (misclassification).

2) ACCURACY ON UNKNOWN SAMPLES (AUS) [23]

This is the accuracy in correctly classifying as unknown the images from unknown camera models.

3) NORMALIZED ACCURACY (NA) [23]

This is the average between Accuracy on Known Samples (AKS) and Accuracy on Unknown Samples (AUS) and provides an overall view of a classifier performance in terms of both open- and closed-set scenarios.

4) DETECTION ACCURACY (DA) [21]

This averages the percentage of images from known cameras detected as coming from known models, and the percentage of images from unknown cameras detected as coming from unknown models. This metric does not take into account whether images from known cameras are misclassified to the wrong camera model.

Concerning f-measure, an additional comment is in order. Traditionally, f-measure is defined in terms of precision and recall as

$$\text{f-measure} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.$$

Depending on the definitions of precision and recall employed, we obtain different f-measure definitions. Mendes Júnior et al. [23] has pointed out that it might be inappropriate to consider the unknown classes c_0 as any other known class in terms of true positive (TP), false positive (FP), and false negative (FN) calculations. Therefore, considering n the number of known camera models, and the $(n + 1)$-th class concerning the unknown classes c_0, we resort to the following f-measure definitions:

5) OPEN-SET MACRO-AVERAGING F-MEASURE (OSFM$_M$ [23]

F-measure using precision and recall defined as

$$\text{precision} = \frac{\sum_{i=1}^{n} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}}{n},$$

$$\text{recall} = \frac{\sum_{i=1}^{n} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}}{n}. \qquad (1)$$

6) OPEN-SET MICRO-AVERAGING F-MEASURE (OSFM$_\mu$) [23]

F-measure using precision and recall defined as

$$\text{precision} = \frac{\sum_{i=1}^{n} \text{TP}_i}{\sum_{i=1}^{n} (\text{TP}_i + \text{FP}_i)},$$

$$\text{recall} = \frac{\sum_{i=1}^{n} \text{TP}_i}{\sum_{i=1}^{n} (\text{TP}_i + \text{FN}_i)}. \qquad (2)$$

7) TRADITIONAL BINARY-BASED MACRO-AVERAGING F-MEASURE(FM$_M$) [38]

F-measure using precision and recall defined as

$$\text{precision} = \frac{\sum_{i=1}^{n+1} \frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}}{n + 1},$$

$$\text{recall} = \frac{\sum_{i=1}^{n+1} \frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}}{n + 1}.$$

8) TRADITIONAL BINARY-BASED MICRO-AVERAGING F-MEASURE(FM$_\mu$) [38]

F-measure using precision and recall defined as

$$\text{precision} = \frac{\sum_{i=1}^{n+1} \text{TP}_i}{\sum_{i=1}^{n+1} (\text{TP}_i + \text{FP}_i)},$$

$$\text{recall} = \frac{\sum_{i=1}^{n+1} \text{TP}_i}{\sum_{i=1}^{n+1} (\text{TP}_i + \text{FN}_i)}.$$

The main difference between traditional and open-set versions of f-measure is that the latter does not consider the effect of the unknown class in terms of TP as the unknown cannot represent a single positive class. Indeed, the sum index spans the range $[1, n]$ rather than $[1, n+1]$, thus excluding the label c_0 representing the unknown classes. However, both OSFM$_M$ and OSFM$_\mu$ account for false known and false unknown through FP and FN, respectively, in Equations (1) and (2).

V. RESULTS

We have evaluated all combinations of extracted features (i.e., 5), training protocols (i.e., 3), and classifiers (i.e., 12) for a total amount of 180 cases of study. Results for each metric are reported in a complete and detailed table of all our experiments, as a supplementary material.[4]

Results show that, overall, better performance are obtained for PISVM, ET, and SSVM classifiers. Regarding the training protocols, interestingly, Open has presented slightly superior results compared to NetOpen, despite using less known-unknown data. And, finally, $\mathbf{f}_{ip1}$ presents the better

---

[4]See supplementary material available at https://pedrormjunior. github.io/oscmi.html.

TABLE 1. Best results in terms of Normalized Accuracy (NA) achieved with each feature extractor. For each metric, the highest results are reported in bold and the lowest ones are reported in italics.

Feature	Classifier	Training Protocol	Best NA	AKS	AUS	DA	OSFM$_M$	OSFM$_\mu$	FM$_M$	FM$_\mu$
f_{ip1}	PISVM	Open	**0.8270**	**0.8639**	*0.7902*	0.8111	0.6916	**0.6410**	0.7042	0.8055
f_{ip2}	ET	Open	0.8189	0.8581	0.7797	*0.7998*	**0.7288**	0.6306	**0.7377**	*0.7959*
f_{conv}	PISVM	NetOpen	0.7779	0.6875	0.8683	**0.8347**	0.6633	0.6220	0.6754	**0.8309**
f_{cfa}	SSVM	NetOpen	0.6825	0.4650	0.9001	0.8112	0.3565	0.5019	0.3849	0.8101
f_{rich}	SVM	Open	*0.5769*	*0.1670*	**0.9868**	0.8175	*0.0844*	*0.2741*	*0.1312*	0.8172

TABLE 2. Best results in terms of NA achieved with each training protocol. For each metric, the highest results are reported in bold and the lowest ones are reported in italics.

Training Protocol	Feature	Classifier	Best NA	AKS	AUS	DA	OSFM$_M$	OSFM$_\mu$	FM$_M$	FM$_\mu$
Open	f_{ip1}	PISVM	**0.8270**	0.8639	0.7902	0.8111	**0.6916**	**0.6410**	**0.7042**	0.8055
Closed	f_{ip1}	SSVM	0.7993	**0.8737**	*0.7250*	*0.7623*	*0.6275*	*0.5904*	0.6453	*0.7557*
NetOpen	f_{ip2}	SOFTMAX	*0.7847*	*0.7205*	**0.8489**	**0.8224**	0.6276	0.6265	*0.6410*	**0.8223**

result among the features, although f_{ip2}, in general, seems to be the most discriminative one.[4] Hereinafter we report a subset of the obtained results in order to highlight the most interesting findings in terms of best feature set, training protocol, and classifier.

A. FEATURE EXTRACTORS
To identify the feature vector most suitable for open-set camera model identification problem, we analyzed the behavior of all features (i.e., f_{rich}, f_{cfa}, f_{conv}, f_{ip1}, and f_{ip2}) paired with different training strategies and classifiers. To summarize the achieved results, we rely on NA as preferred analysis metric. As a matter of fact, NA clearly takes into account the ability of correctly classifying known samples at camera level as well as rejecting the unknown. Therefore, an algorithm with high NA value is a good candidate to work for both known and unknown classes.

Table 1 reports the best NA achieved with each feature extractor. Specifically, it shows which combination of classifier and training strategy enables to obtain the achieved NA, as well as all the other metric values for the selected classifier. From this table, it is possible to notice that the best results are obtained by CNN-based features. In particular, f_{ip1} achieves the best NA, which is close to 0.83. This confirms the behavior observed by Bondi *et al.* [15] for the closed-set scenario: hand-crafted features (i.e., f_{rich} and f_{cfa}) performs better on high resolution images, whereas the CNN is superior when trained on small 64×64 pixel patches as the ones considered in this work. The explanation for the affected accuracy with hand-crafted features when working with small patches is that hand-crafted features relies on co-occurrences [11], [25], whose computation for small patches might be less stable and reliable.

It is interesting to notice how AKS and AUS are unbalanced for hand-crafted features. For instance, f_{rich} and f_{cfa} show AUS higher than 0.90, but AKS lower than 0.50. This means that the classifier rejects many more images as unknown than it should. This makes these features not appealing for open-set problems, as the presence of unknown devices greatly hinders the closed-set classification capability

of these features. The same behavior is also captured by the metrics based on f-measure. Conversely, f_{ip1} is able to correctly classify unknown images with almost 0.80 accuracy (AUS), and to correctly attribute known-camera images to their model with 0.86 accuracy (AKS).

B. TRAINING PROTOCOLS
To evaluate the different training protocols, we considered NA as reference metric for the same reasons previously mentioned. Table 2 reports the best NA results for each protocol, also showing which feature and classifier is used to obtain the reported result. Also, the other metrics are then reported for each case.

It is possible to notice that Open strategy presents better results, more than 4% higher than the best result with NetOpen. In Table 2, although Closed strategy presents better results than NetOpen, in general, we have observed that Closed tends to perform the worse.[4] Also in a general evaluation, we also observe that, in fact, Open tends to perform slightly better than NetOpen.

It is worth to highlight one aspect about the Closed strategy. Despite this strategy's name, all classifiers employed along with it are open-set ones. Therefore, even if trained only considering known camera images, they still have the ability to reject new data as unknown (remember, from Section III-C, the different training protocols refers only to the split of the training data). This explains why using the Closed strategy is still possible to achieve AUS higher than 0.70. However, even though, Open approaches 0.80, almost 10% of difference from the Closed strategy is observed.

Furthermore, considering all the 360 measured combinations (12 classifiers × 5 features sets × 6 metrics), classifiers training with Open obtained better results than versions trained with NetOpen in 188 of the them, while NetOpen wins in 172 cases. It also indicates a slightly better performance for Open protocol. However, when NetOpen achieves better results, the classifiers obtain an average of about 10.5% better results, while Open improves only 7.8% in average.

This is a counter intuitive result, as NetOpen uses the same known data as Open strategy does, along with

TABLE 3. Best results in terms of NA achieved with each open-set classifier. For each metric, the highest results are reported in bold and the lowest ones are reported in italics.

Classifier	Feature	Training Protocol	Best NA	AKS	AUS	DA	OSFM$_\text{M}$	OSFM$_\mu$	FM$_\text{M}$	FM$_\mu$
PISVM	f_{ip1}	Open	**0.8270**	0.8639	0.7902	0.8111	0.6916	0.6410	0.7042	0.8055
ET	f_{ip2}	Open	0.8189	0.8581	0.7797	0.7998	**0.7288**	0.6306	**0.7377**	0.7959
SSVM	f_{ip1}	Closed	0.7993	**0.8737**	0.7250	0.7623	0.6275	0.5904	0.6453	0.7557
SOFTMAX	f_{ip2}	NetOpen	0.7847	0.7205	0.8489	0.8224	0.6276	0.6265	0.6410	0.8223
OSNN	f_{ip2}	NetOpen	0.7841	0.6813	0.8869	0.8444	0.6344	**0.6441**	0.6485	**0.8443**
SVM	f_{ip1}	Open	0.7626	0.7575	0.7676	0.7676	0.5305	0.5698	0.5514	0.7655
PSVM	f_{conv}	Open	0.7544	0.7151	0.7938	0.7792	0.6033	0.5689	0.6180	0.7775
NCM	f_{ip2}	Open	0.7339	0.8530	0.6149	0.6722	0.6379	0.5064	0.6514	0.6641
OCSVM	f_{ip2}	Open	0.6742	0.3873	0.9611	**0.8471**	0.5048	0.4967	0.5333	0.8424
DBC	f_{conv}	Open	0.6371	0.7902	*0.4840*	*0.5535*	0.5128	0.4161	0.5335	*0.5474*
2PSVM	f_{cfa}	Closed	0.5881	0.3844	0.7917	0.7130	0.3615	0.3479	0.3854	0.7075
WSVM	f_{cfa}	Open	*0.5079*	*0.0287*	**0.9870**	0.7953	*0.0748*	*0.0517*	*0.1295*	0.7888

extra known-unknown data from the other Dresden Image Database classes not employed as known. The numbers regarding the difference of those two training protocols indicates some similarity among the representativeness of the two sets of training data. Therefore, those results indicate that by simply having some known-unknown data, although they are not unknown from the point of view of the network (Open strategy), is enough for improving the performance compared to the traditional Closed form. It means those extra data are not necessary, which is a good trace also for making the training process cheaper.

Moreover, those results are good evidences that representation of unknown instances are as distinct as the representations of known-unknown from the point of view of the network. It means those representations are distinct alike from the known instances after a trained network is employed for feature extraction. Those results are also in tune with the ones presented by Bondi *et al.* [15]: they have performed a closed-set experiment with a distinct set of camera models not employed for network training and they have showed that representations for those camera models are distinct enough to allow discrimination among them.

C. OPEN-SET CLASSIFIERS
To analyze the effect of different classifiers, Table 3 reports the best NA result obtained with each classifier, showing also the feature and training strategy used in each case. For each selected methodology, all other metrics are also reported.

From these results—as we saw in other tables as well—it is possible to see that PISVM performs better than its counterparts, achieving NA close to 0.83, however, best AKS and AUS are obtained with SSVM and OCSVM, respectively. Results in Table 3 show many classifiers with reasonable performance: among the cases, ET have obtained the best performance for the macro-averaging versions of the f-measure measures and OSNN presents best results for the micro-averaging versions. OCSVM also outperforms other methods based on DA although its high propensity of rejecting instances as unknown. Additionally, Closed protocol only appears to be the best one for SSVM and 2PSVM classifiers, all other classifiers has the Open or NetOpen

variations as the best training protocol, and Open appears in most of the cases.

It is important to notice that 2PSVM appears as one of the last methods in the ranking of Table 3. This low performance for 2PSVM can be justified by its implicit assumption that all known classes can be modeled as a single class. It does not take into account the fact that known classes can be sparse in the feature space and some intermediate regions among those classes can refer to the unknown, i.e., it is difficult to specialize on the known classes by means of a single model. Furthermore, the best NA result with 2PSVM is obtained with Closed training protocol, which indicates that even though simulation of the open-set scenario is performed for parameter optimization, a one-class classifier is not able to handle well the feature space.

In general, we verify that by the straightforward employment of an open-set classifier, as is, improves results for the open-set scenario compared to closed-set classifiers adapted for open-set recognition by means rejection through thresholding of similarity scores. Further details regarding comparison with those state-of-the-art solutions are presented in the next section.

D. COMPARISON WITH STATE-OF-THE-ART
To the best of our knowledge, the only work presenting results for the open-set camera model identification problem is the work of Bayar and Stamm [21]. In particular, in this work, the authors propose two different approaches. The first one (V-D.1) relies on confidence score thresholding: when the classifier is not "sure" about its classification to a certain known class, test instance is then rejected as unknown. The second approach (V-D.2) assumes known-unknown data is available for training a classifier for detecting if a test instance is known or unknown. For this approach, previous work have evaluated only the detection ability although in a real open-set scenario further decision should be required to chose the correct class in case an instance is detected as known.

1) APPROACH 1
The first approach proposed by Bayar and Stamm [21] works as it follows. A multi-class classifier is trained with Closed

TABLE 4. Difference achieved by the best solution found through the pipeline considered in this work (PISVM) and the baselines. Results obtained for f_{ip1} feature for the corresponding methods implemented along with Open training protocol. The consistency of positive values for Δ evinces the improvement of the found solution over the state-of-the-art methods.

Reference	ΔNA	ΔAKS	ΔAUS	ΔDA	ΔOSFM$_M$	ΔOSFM$_\mu$	ΔFM$_M$	ΔFM$_\mu$
SOFTMAX	0.3270	0.8639	*−0.2098*	0.0179	0.6916	0.6410	0.6577	0.0123
ET	0.0547	0.1641	*−0.0546*	*−0.0044*	0.0204	0.0319	0.0221	*−0.0094*
NCM	0.1698	0.0748	0.2649	0.2205	0.1375	0.2099	0.1323	0.2256
PSVM	0.0938	0.1205	0.0670	0.0770	0.0513	0.1169	0.0512	0.0781

training protocol.[5] This classifier is chosen in order to also provide a confidence score about detected class. Instances providing a low confidence score are classified as unknown. For this class of methods, we implemented their solutions based on thresholding softmax probability (SOFTMAX), Nearest Class Mean with cosine distance (NCM), SVM with Platt's probability for rejection (PSVM), and ET [36].

Table 4 reports the metric difference Δ achieved by the best solution we have evaluated in previous sections compared to the baselines, by considering, for each method, the setup that maximizes NA. From this comparison, in general, it is possible to notice that the best solution we found in our analysis is able to achieve better results than all strategies reported by Bayar and Stamm [21]. For most of the measures, for each of the compared baselines, PISVM improves the accuracy.

We see in the same table that ET, as employed by Bayar and Stamm [21], is the most competitive method compared to a classifier specially designed for open-set scenario (PISVM). Although its high accuracy, we should analyze some theoretical properties of the classifier. For instance, consider the ability of bounding the region of the feature space in which a possible test instance would be classified as belonging to one of the known classes, i.e., bounding the known-labeled open space (KLOS) [23], [24]. Figures 5(a) and 5(b) depict the decision boundaries of PISVM and ET classifiers, respectively, in the feature space formed by the two first features of the f_{ip2} layer. For those images, only training samples from the 4 first classes, out of the 18, were employed to avoid cluttering the visualization. Small circles represent training samples. Colored regions indicate that a possible test instance in there would be classified to the class of the same color. The white region represents rejection as unknown. In Figure 5, we observe that PISVM is able to bound the KLOS, properly ensuring the rejection of any data point that would appear far from the support of the training samples in the feature space. However, by thresholding the probability score of the ET classifier, the same property is not ensured. In general, we see that PISVM demonstrates a more controlled behavior.

2) APPROACH 2

The second approach proposed by Bayar and Stamm [21] works as it follows. A binary classifier is trained to

FIGURE 5. Decision boundaries of the (a) PISVM open-set classifier compared to competing (b) ET classifier.

distinguish between images from known and unknown camera models. The objective here is to analyze only the detection ability. All samples from all known classes are considered into a single known class called known. Extra data from other classes not of interest are employed on the unknown class for the binary classification. As in the previous experiments, we consider the 18 classes of Dresden as the known classes of interest. For the extra known-unknown data, we employed the remaining classes of the Dresden dataset, as those classes were also employed along with NetOpen training protocol. For this method, we implemented both solutions shown by Bayar and Stamm [21], i.e., PSVM[6] and ET.

In Table 5, we present the DA for the two baselines as well as for PISVM solution which have presented the best results throughout our experiments. Furthermore, Detection on Known Samples (DKS) and Detection on Unknown Samples (DUS) are also presented for a more in-depth evaluation of the performance of the classifiers. NetOpen was selected for those results because baselines require extra known-unknown data for training although PISVM have obtained best performance along with Open strategy (Section V-C). f_{ip2} is employed in this case because, as previously saw[4], it has comparable or better results than f_{ip1} in general and, furthermore, baselines has presented slightly better results with this feature, compared to f_{ip1}.

Our results for this approach, as seen in Table 5, are far from the ones reported by Bayar and Stamm [21] as the baselines have almost no ability to reject instances as unknown. Our conclusions from those results is that by relying solely on known-unknown data for training a classifier

---

[5]Previous work [21] have not evaluated neither Open nor NetOpen protocols. To the best of our knowledge, our work evaluates them for the first time in this problem.
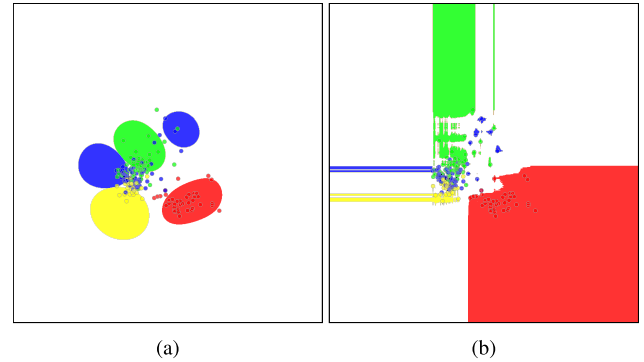
[6]Notice, however, that Platt's probability is not required to be employed in this context as only the class decision matters in this case.

TABLE 5. Difference achieved by the best solution found through the pipeline considered in this work (PISVM) and the baselines considering Approach 2 of Bayar and Stamm [21]. Results obtained for f_{ip2} feature for the corresponding methods implemented along with NetOpen training protocol. For each metric, the highest results are reported in bold.

Reference	DA	DKS	DUS
PISVM	**0.7419**	0.8806	**0.7057**
PSVM	0.2069	**1.0000**	0.0000
ET	0.2087	**1.0000**	0.0024

to distinguish, in the wild, known versus unknown classes is susceptible to a worst case scenario. We conjecture that the known-unknown data employed for those classifiers makes them create a decision frontier in the feature space in such a way that most of the real unknown data (from Image Source Attribution Unicamp (ISA Unicamp) and Flickr Unicamp datasets) becomes accepted as known. If a different set of known-unknown is employed in place of the unknown part of the Dresden dataset, we believe results might drastically differ. Taking an essentially distinct approach, PISVM along with NetOpen training protocol does not rely solely on the known-unknown data for defining its boundary decision: instead, it minimizes the risk of the unknown also taking advantage of the inter-class information gathered from the known data [22].

E. POST-FUSION ANALYSIS

In the machine learning field, it is well known that jointly using a series of different models can help increasing classification performance. This is known as ensemble learning [39]. In the light of this, here we present results achieved with a very simple yet effective ensemble fusion technique. We perform majority voting among different models. Given a set of trained models, we test the image under analysis with all of them, and perform majority voting on their output. If the majority of the votes is for rejecting as unknown, the image is then classified as unknown.

By considering all 1048575 combinations obtained by fusing up to 8 single models achieving NA greater than 0.7. Top-three results are reported in Table 6. Notice that, the features that are selected are always f_{ip1} and f_{ip2}. Moreover, top results includes all three training protocols. The classifiers that appear among those selected solutions are PISVM, SSVM, OSNN, and ET. These results confirm that by using post-fusion it is actually possible to increase NA of approximately 2.5%, and no more than 6 models are needed. This paves the way to the development of more complex ensemble methods for open-set camera model identification.

F. IMPACT OF AN OPEN-SET SOLUTION

In Figure 6, we present two confusion matrices. One of them, in Figure 6(a), obtained by an open-set solution and the other, in Figure 6(b), by the closed-set output of the neural network employed along this work. By comparing Figures 6(a) and 6(b), we observe that the ability on recognizing instances of each individual model is affected on
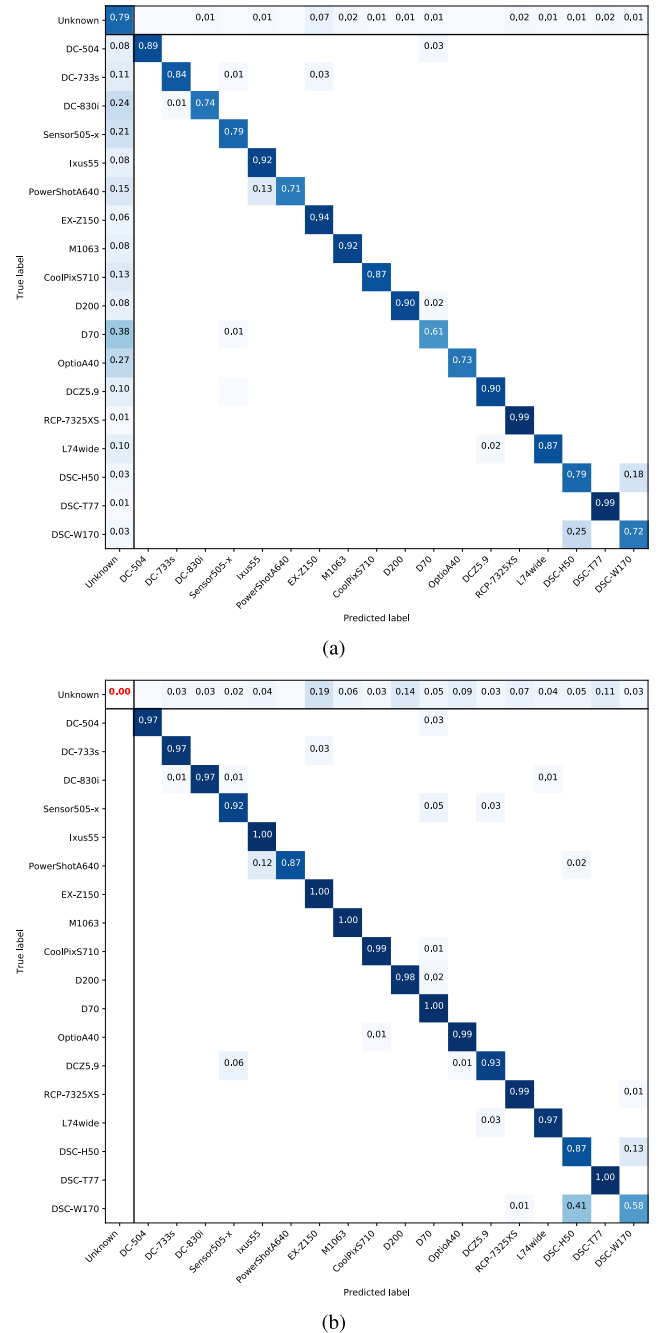

(a)

(b)

FIGURE 6. Confusion matrix comparing the best open-set results against the baseline closed-set CNN solution [15], that is equivalent to the SOFTMAX method [21] without the rejection threshold. (a) Open-set solution with PISVM on f_{ip1} features along with Open training protocol. (b) Closed-set solution as the output of the network, i.e., without reject option.

the open-set solution. That is expect as long as an open-set solution can also perform the fault of rejecting instances as unknown, i.e., false unknown, while a closed-set solution can only have misclassifications. On the other hand, we clearly see the undesirable behavior of the closed-set solution of assigning every unknown instance to one of the known models, i.e., 0% on AUS or, in another perspective, 100% of false known. The false known rate obtained by the open-set solution, in this example, is 21%. Anyhow, it is worth noticing

TABLE 6. Top three post-fusion results in terms of NA by considering the alternatives with accuracy greater than 0.7 and the fusion of at most 8 models.

Combination	N. models	NA
(Open, OSNN, f_{ip2}), (NetOpen, PISVM, f_{ip2}), (Open, SSVM, f_{ip2}), (Closed, SSVM, f_{ip1}), (Open, ET, f_{ip2}), (Open, PISVM, f_{ip1})	6.0000	0.8522
(NetOpen, SSVM, f_{ip1}), (Open, OSNN, f_{ip2}), (Open, SSVM, f_{ip2}), (Closed, SSVM, f_{ip1}), (Open, ET, f_{ip2}), (Open, PISVM, f_{ip1})	6.0000	0.8521
(Open, OSNN, f_{ip2}), (Open, SSVM, f_{ip2}), (Closed, SSVM, f_{ip1}), (Open, ET, f_{ip2}), (Open, PISVM, f_{ip1})	5.0000	0.8519

that most of the open-set classifiers can be tuned to decrease its false known rate although with the expense of increasing its false unknown rate.

VI. CONCLUSIONS

In this paper, we studied the use of a supervised-learning strategy for image camera model identification in an open-set scenario. In doing so, we explored the possibility of using multiple camera-related features proposed in the literature for closed-set camera model identification, however, under the more challenging open-set regime. We considered pairing feature vectors with different open-set classifiers exploring also the use of three alternatives of training protocols. All tests have been performed considering a selection of three independent image datasets freely available online comprising a large number of images from more than 300 camera models.

In terms of training protocols, we found out that employing extra known-unknown classes, as for NetOpen approach, in general does not help on improving the performance of the classifiers compared to the simpler and cheaper employment of the Open strategy. This result is interesting as it evinces that extra known-unknown classes, from the point of view of the network, are not required to be employed as its impact is limited. It means one can successfully train any open-set classifier, along with an Open training protocol, with only the data available for the known classes. A better intuition on this behavior requires a deeper study on the network's representation for unknown classes not employed on network training and those should be compared among the representation of each of the known classes employed for training the network, therefore, it remains as a future work.

Another evidence on the limited use of the known-unknown data from the point of view of the network were presented by employing a binary classifier for recognizing known versus unknown camera models: when a known-unknown set of data (from the unknown part of Dresden) is employed to train this classifier, its performance on detecting unknown camera models from ISA Unicamp and Flickr datasets is highly effected (Section V-D.2). It also reinforces previous arguments on the open-set area that more theoretically-sounded and less data-relied solutions should be developed for general open-set problems [24].

Our results have shown that appropriate means of dealing with the open-set camera model attribution problem should be sought in order to properly handling the problem, considering that a recently proposed open-set method [22], as is, obtains considerable improved results compared to the straightforward idea of thresholding the softmax probability of neural networks for rejection as unknown (Section V-D.1). This

problem on thresholding the softmax probability for open-set recognition have been evinced in one of our previous work, hence the current work also confirms the previously more theoretical perspective [35, Chapter 7].

For the open-set camera model identification problem, a promising future research can be performed on investigating recently proposed alternatives to the softmax loss, e.g., the center loss [40], the angular softmax loss [41], etc., as the authors of those works have claimed improvement on the open-set face recognition problem.

REFERENCES

[1] M. C. Stamm, M. Wu, and K. J. R. Liu, "Information forensics: An overview of the first decade," *IEEE Access*, vol. 1, pp. 167–200, May 2013.

[2] A. Piva, "An overview on image forensics," *ISRN Signal Process.*, vol. 2013, pp. 1–22, Oct. 2012.

[3] A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein, "Vision of the unseen: Current trends and challenges in digital image and video forensics," *ACM Comput. Surv.*, vol. 43, no. 4, p. 26, 2011.

[4] M. Kirchner and T. Gloe, "Forensic camera model identification," in *Handbook of Digital Forensics of Multimedia Data and Devices*. Chichester, U.K.: Wiley, 2015, pp. 329–374.

[5] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, "Source camera identification based on CFA interpolation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2005, p. III-69.

[6] G. Cao, Y. Zhao, R. Ni, L. Yu, and H. Tian, "Forensic detection of median filtering in digital images," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2010, pp. 89–94.

[7] X. Zhao and M. C. Stamm, "Computationally efficient demosaicing filter estimation for forensic camera model identification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 151–155.

[8] S.-H. Chen and C.-T. Hsu, "Source camera identification based on camera gain histogram," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep./Oct. 2007, pp. IV-429–IV-432.

[9] K. S. Choi, E. Y. Lam, and K. Y. Wong, "Automatic source camera identification using the intrinsic lens radial distortion," *Opt. Express*, vol. 14, pp. 11551–11565, 2006.

[10] T. H. Thai, R. Cogranne, and F. Retraint, "Camera model identification based on the heteroscedastic noise model," *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 250–263, Jan. 2014.

[11] C. Chen and M. C. Stamm, "Camera model identification framework using an ensemble of demosaicing features," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Nov. 2015, pp. 1–6.

[12] F. Marra, G. Poggi, C. Sansone, and L. Verdoliva, "Evaluation of residual-based local features for camera model identification," in *Proc. Int. Conf. Image Anal. Process.* Cham, Switzerland: Springer, 2015, pp. 11–18.

[13] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification based machine learning approach with high order statistics features," in *Proc. Eur. Signal Process. Conf. (EUSIPCO)*, Aug./Sep. 2016, pp. 1183–1187.

[14] A. Tuama, F. Comby, and M. Chaumont, "Camera model identification with the use of deep convolutional neural networks," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2016, pp. 1–6.

[15] L. Bondi, L. Baroffio, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "First steps toward camera model identification with convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 259–263, Mar. 2017.

[16] L. Bondi, D. Güera, L. Baroffio, P. Bestagini, E. J. Delp, and S. Tubaro, "A preliminary study on convolutional neural networks for camera model identification," *Electron. Imag.*, vol. 2017, no. 7, pp. 67–76, 2017.

[17] A. Ferreira, H. Chen, B. Li, and J. Huang, "An inception-based data-driven ensemble approach to camera model identification," in *Proc. IEEE Int. Workshop Inf. Forensics Secur.*, Hong Kong, Dec. 2018, pp. 1–7.

[18] B. Bayar and M. C. Stamm, "Augmented convolutional feature maps for robust CNN-based camera model identification," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Beijing, China Sep. 2017, pp. 4098–4102.

[19] L. Bondi, S. Lameri, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro, "Tampering detection and localization through clustering of camera-based CNN features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1855–1864.

[20] T. Gloe, "Feature-based forensic camera model identification," in *Transactions on Data Hiding and Multimedia Security VIII* (Lecture Notes in Computer Science), vol. 7228. Berlin, Germany: Springer, 2012, pp. 42–62.

[21] B. Bayar and M. C. Stamm, "Towards open set camera model identification using a deep learning framework," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2007–2011.

[22] L. P. Jain, W. J. Scheirer, and T. E. Boult, "Multi-class open set recognition using probability of inclusion," in *Computer Vision—ECCV*. Cham, Switzerland: Springer, 2014, pp. 393–409.

[23] P. R. Mendes Júnior, R. M. de Souza, R. de O. Werneck, B. V. Stein, D. V. Pazinato, W. R. de Almeida, O. A. B. Penatti, R. da S. Torres, and A. Rocha, "Nearest neighbors distance ratio open-set classifier," *Mach. Learn.*, vol. 106, pp. 359–386, Mar. 2017.

[24] W. J. Scheirer, A. D. R. Rocha, A. Sapkota, and T. E. Boult, "Towards open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1757–1772, Jul. 2013.

[25] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 868–882, Jun. 2012.

[26] D. Cozzolino, D. Gragnaniello, and L. Verdoliva, "Image forgery localization through the fusion of camera-based, feature-based and pixel-based techniques," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 5302–5306.

[27] G. Xu and Y. Q. Shi, "Camera model identification using local binary patterns," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2012, pp. 392–397.

[28] O. Çeliktutan, B. Sankur, and I. Avcibaş, "Blind identification of source cell-phone model," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 553–566, Sep. 2008.

[29] A. Rocha and S. K. Goldenstein, "Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 289–302, Feb. 2014.

[30] B. Heflin, W. Scheirer, and T. E. Boult, "Detecting and classifying scars, marks, and tattoos found in the wild," in *Proc. IEEE Int. Conf. Biometrics, Theory, Appl. Syst.*, Arlington, VA, USA, Sep. 2012, pp. 31–38.

[31] D. A. Pritsos and E. Stamatatos, "Open-set classification for automated genre identification," in *Proc. Eur. Conf. Inf. Retr.*, in Lecture Notes in Computer Science, vol. 7814, P. Serdyukov, P. Braslavski, S. O. Kuznetsov, J. Kamps, and S. Rüger, Eds. Moscow, Russia: Springer, Mar. 2013, pp. 207–217.

[32] W. J. Scheirer, L. P. Jain, and T. E. Boult, "Probability models for open set recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2317–2324, Nov. 2014.

[33] F. de O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha, "Open set source camera attribution," in *Proc. SIBGRAPI Conf. Graph., Patterns Images (SIBGRAPI)*, Aug. 2012, pp. 71–78.

[34] F. de O. Costa, E. Silva, M. Eckmann, W. J. Scheirer, and A. Rocha, "Open set source camera attribution and device linking," *Pattern Recognit. Lett.*, vol. 39, pp. 92–101, Apr. 2014.

[35] P. R. Mendes Júnior, "Open-set recognition for different classifiers," Ph.D. dissertation, Inst. Comput., Univ. Campinas, Campinas, Brazil, Sep. 2018. [Online]. Available: http://repositorio.unicamp.br/handle/REPOSIP/333051

[36] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Mach. Learn.*, vol. 63, no. 1, pp. 3–42, 2006.

[37] T. Gloe and R. Böhme, "The Dresden image database for benchmarking digital image forensics," *J. Digit. Forensic Pract.*, vol. 3, nos. 2–4, pp. 150–159, 2010.

[38] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.

[39] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 4, p. e1249, Jul./Aug. 2018.

[40] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A comprehensive study on center loss for deep face recognition," *Int. J. Comput. Vis.*, vol. 127, nos. 6–7, pp. 668–683, 2019.

[41] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jun. 2017, pp. 212–220.

PEDRO RIBEIRO MENDES JÚNIOR is a Brazilian. He received the B.Sc. degree in computer science from the Federal University of Ouro Preto (UFOP), Ouro Preto, Minas Gerais, Brazil, in 2012, and the M.Sc. and Ph.D. degrees in computer science from the University of Campinas (UNICAMP), Campinas, São Paulo, Brazil, in 2014 and 2018, respectively. He is currently a Postdoctoral Fellow of the Image and Sound Processing Group, Politecnico di Milano, Milan, Italy, where he is currently involved on the DeepEyes Project supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) from Brazil. His research interests include open-set recognition, and currently, open-set methods and techniques applied to multimedia forensics applications.

LUCA BONDI received the M.Sc. degree in computer science and engineering and the Ph.D. degree in information technology from the Politecnico di Milano, Milan, Italy, in 2014 and 2019, respectively, where he is currently a Postdoctoral Fellow of the Image and Sound Processing Group. His research interest includes data-driven methods applied to images and videos in multimedia forensics applications.

PAOLO BESTAGINI was born in Novara, Italy, in 1986. He received the M.Sc. degree in telecommunications engineering and the Ph.D. degree in information technology from the Politecnico di Milano, Italy, in 2010 and 2014, respectively, where he is currently an Assistant Professor with the Image and Sound Processing Group. His research interest includes multimedia forensics and acoustic signal processing for microphone arrays. He is an Elected Member of the IEEE Information Forensics and Security Technical Committee and the Co-Organizer of the IEEE Signal Processing Cup 2018.

STEFANO TUBARO was born in Novara, Italy, in 1957. He received the degree in electronic engineering from the Politecnico di Milano, Milan, Italy, in 1982. He then joined the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, first as a Researcher of the National Research Council, and then an Associate Professor, in 1991. Since 2004, he has been a Full Professor of telecommunication with the Politecnico di Milano. His current research interest includes advanced algorithms for video and sound processing. He is the author of more than 180 scientific publications in international journals and congresses He holds over 15 patents. In the past few years, he has focused his interest on the development of innovative techniques for image and video tampering detection, in general, for the blind recovery of the processing historyÌ of multimedia objects. He coordinates the research activities with the Image and Sound Processing Group, Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano. He had the role of Project Coordinator of the European Project ORIGAMI: A new paradigm for high-quality mixing of real and virtual, and the research project ICT-FET-OPEN REWIND: REVerse engineering of audio-VIsual coNtent Data. This last project was aimed at synergistically combining principles of signal processing, machine learning, and information theory to answer relevant questions on the past history of such objects. He is a member of the IEEE Multimedia Signal Processing Technical Committee and the IEEE SPS Image Video and Multidimensional Signal Technical Committee. He was on the Organization Committee of a number of international conferences including the IEEE MMSP 2004/2013, the IEEE ICIP 2005, the IEEE AVSS 2005/2009, the IEEE ICDSC 2009, the IEEE MMSP 2013, and the IEEE ICME 2015. From 2012 to 2015, he was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is currently an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.

ANDERSON ROCHA is currently an Associate Professor with the Institute of Computing, University of Campinas (UNICAMP), Brazil. Since 2019, he has been the Director of the Institute of Computing, University of Campinas (UNICAMP). His main interests include digital forensics, reasoning for complex data, and machine intelligence. He has actively involved as a Program Committee Member of several important computer vision, pattern recognition, and digital forensics events. He is an Elected Affiliate Member of the Brazilian Academy of Sciences (ABC) and the Brazilian Academy of Forensic Sciences (ABC). He is a two-term Elected Member and the Chair of the IEEE Information Forensics and Security Technical Committee (IFS-TC), from 2019 to 2020. He is a Microsoft Research and a Google Research Faculty Fellow, important academic recognitions given to researchers by Microsoft Research and Google, respectively. In addition, he has received the Tan Chin Tuan (TCT) Fellowship, in 2016 and a recognition promoted by the Tan Chin Tuan Foundation in Singapore. He has been the Principal Investigator of a number of research projects in partnership with public funding agencies in Brazil and abroad, and national and multi-national companies having already deposited and licensed several patents. He is a Brazilian CNPq Research Scholar (PQ1). He is currently an Associate Editor of important international journals such as the IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY (TIFS), the *Journal of Visual Communication and Image Representation* (JVCI) (Elsevier), the *Journal on Image and Video Processing* (JIVP) (Springer), and the *IEEE Security and Privacy Magazine*.

● ● ●