

Research Article

Multi-DS Strategy for Source Camera Identification in Few-Shot Sample Data Sets

Bo Wang,¹ Jiayao Hou,¹ Yanyan Ma ,¹ Fei Wang,¹ and Fei Wei ²

¹School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, Liaoning, China

²Department of Electrical Engineering, Arizona State University, Tempe 85281, AZ, USA

Correspondence should be addressed to Fei Wei; fwei16@asu.edu

Received 7 May 2022; Revised 18 July 2022; Accepted 29 July 2022; Published 6 September 2022

Academic Editor: Hanzhou Wu

Copyright © 2022 Bo Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Source camera identification (SCI) is an intriguing problem in digital forensics, which identifies the source device of given images. However, most existing works require sufficient training samples to ensure performance. In this work, we propose a method based on semi-supervised ensemble learning (multi-DS) strategy, which extends labeled data set by multi-distance-based clustering strategy and then calibrate the pseudo-labels through a self-correction mechanism. Next, we iteratively perform the calibration-appending-training process to improve our model. We design comprehensive experiments, and our model achieves satisfactory performance on benchmark public databases (i.e., Dresden, VISION, and SOCRatES).

1. Introduction

Digital images are widely used in social media and people's modern life. An enormous number of user-friendly photo editing apps are available in the market, such that it is important to identify the authenticity of digital images for preventing digital images from being used by intentional people and suffering malicious tampering. In the field of digital image forensics, the most important approach is passive forensics, as it is often difficult to obtain reliable digital watermark or digital label information in the actual judicial forensic scenario. Source camera identification (SCI) is an important branch in the field of digital image forensics [1]. Due to the difference in device types, models, individual hardware, and build-in image generation algorithms in the image-creating process, these unique marks may be left as trace in the images. For example, Lucas et al. [2] analyze the noise introduced in the imaging process and take the photo-response nonuniformity (PRNU) noise as the fingerprint, which proved the feasibility of mining the internal traces of the image.

Previous methods obtain high identification accuracy when training sets are sufficiently large, and insufficient training sets (aka few-shot sets) may significantly impact the

performance. However, the construction of large labeled sample set is time-consuming, laborious, and sometimes impossible. It is interesting to find a method, which gives accurate identification results even when the training samples with known labels are limited, or at least provides a more reliable and convincing source identification result. At present, the typical methods to solve few-shot sample problems are usually data expansion, such as generating virtual samples [3, 4], data enhancement [5–7], and semi-supervised learning [8–10]. We attempt to solve the problem of source camera identification in the case of few-shot samples based on the semi-supervised learning method and try to propose an algorithm model that can be applied to actual judicial forensics and other scenarios to ensure a relatively reliable source camera identification accuracy rate.

In this work, we propose a distance-based semi-supervised ensemble learning (multi-DS) strategy to solve the camera source identification problem under few-shot sample conditions. We first perform classifying by comprehensively comparing multiple distances between unlabeled-labeled sample pairs, then mark unlabeled image samples with pseudo-labels by a sorting algorithm, calibrate the pseudo-labels by a support vector machine (SVM)-based self-correction mechanism, and append the new labeled

sample to the training set. By iteratively repeating the labeling-calibrating-appending-training process, we obtain the final model with stopping conditions.

Our contributions are as follows:

- (i) We propose a distance semi-supervised ensemble learning (multi-DS) strategy, which integrates multiple distance indicators to cluster unlabeled samples based on few-shot labeled samples, so as to make full use of the information of few-shot sample data sets.
- (ii) We mark the samples close to the cluster center with pseudo-labels of the same class as reliable pseudo-label samples and use these samples to expand the few-shot sample set for SVM self-correction training, so as to optimize the model through continuous iteration.
- (iii) We conduct comprehensive experiments on multiple threshold parameters in multi-DS strategy to prove the effectiveness of our model. The experimental results show that our strategy is superior to other existing methods on few-shot sample data sets.

The rest of this article is structured as follows. The second part introduces the field of camera source recognition and related work in few-shot sample scenes. The third part introduces in detail our proposed distance-based semi-supervised ensemble learning (multi-DS) strategy. The fourth part provides some background information about the strategy we used. In the fifth part, we discussed the experimental design and results in detail. A summary of our work can be found in Section 6.

2. Related Works

In this work, we mainly focus on camera model identification. In this section, we review the existing work in this field. In the past two decades, based on different levels of camera source identification, people have gradually carried out SCI research from three directions: equipment-based, model-based, and individual-based. We mainly identify the source according to the model of the source camera.

Model-based SCI can identify the source camera model of a given image, to accurately identify the specific brand and model of the source camera. Since each camera brand has a different model, statistical characteristics are of sufficient importance. Kharrazi et al. [11] and others proposed that through the inherent structural differences in the camera model, the statistical characteristics of the image color correlation and color energy ratio can be used for SCI problems. They use wavelet features [12] and image quality features [13] to collect image information and then build higher-order statistical models of natural images. Support vector machines are then used to discriminate between untouched and adulterated images, so as to complete source camera identification. In addition, they also used more features, including statistical features such as color correlation, color energy ratio, and neighborhood distribution

centroid for identification. Çeliktutan et al. [14] combined three sets of forensic features to measure, including binary similarity metric (BSM), image quality metric (IQM), and high-order wavelet statistics (HOWS), which further proved the feasibility of comprehensive forensics of multiple features. Swaminathan et al. [15] searched for the correlation between image pixels caused by color filter array (CFA) interpolation features and proposed a linear interpolation model based on the peaks existing in the spectral domain to estimate the neighborhood CFA interpolation coefficients. Xu et al. [16] proposed a method of combining the local binary pattern (LBP) and local phase quantization (LPQ) features of the hue and value color channels in the hue, saturation, and value (HSV) color space. They extracted the LBP features and the LPQ features from the contourlet transform coefficients of the original image and residual noise image. The LPQ feature recognizes 10 camera brands from the Dresden image data set with an accuracy rate of 99.8%.

However, the work of the above researchers is aimed at the training environment with sufficient labeled samples, but when the labeled training samples are insufficient, the SVM classifier or other learning algorithms cannot be fully trained, which will significantly reduce the classification accuracy of SCI. Tan et al. [17] constructed multiple prototype sets using the ensemble projection (EP) method and used richer features to solve the problem of insufficient known labels in the few-shot sample problem. Liang et al. [18] used the attention multisource fusion few-shot learning method (AMF-FSL) to transfer the classification ability of few-shot learning from multisource data to target data, which improved the generalization ability of the classification model in cross-domain. Sameer and Naskar [19] used the deep Siamese network method to enhance the training space by forming paired samples from the same camera model and different camera models and obtained a better model of camera source identification problem. Huo et al. [20] focused on the scene of zero sample and few sample mixed learning with extreme scarcity. They chose to obtain more noisy label samples from the image search engine for data expansion and then used the enhanced but noisy label training data for projection learning through noise suppression and semantic projection learning algorithms, which provide a feasible scheme for practical application scenarios when data are particularly inadequate.

In fact, the number of training samples in source camera identification is often difficult to meet the needs of identification. Therefore, considering the practical application scenarios of few-shot labeled samples, we propose our strategy below.

3. The Proposed Multi-DS Strategy

To make full use of more detailed information of the few-shot samples, this section proposes a strategy of few-shot sample image source identification based on distance semi-supervised ensemble learning, named multi-DS algorithm.

The complete multi-DS algorithm block diagram is shown in Figure 1. After extracting sufficient features for all

data sets, two screenings are performed based on the distance threshold and the statistical frequency threshold based on the three distance indicators to complete the semi-supervised ensemble learning process, and the pseudo-label information is corrected repeatedly until it is stable. The details of each part of the algorithm are as follows.

The block diagram of feature extraction and multi-distance integrated integration to filter pseudo-label samples is shown in Figure 2. To fully obtain the detailed information of all camera images, multi-statistical features are extracted for all samples in training and testing. At the same time, to ensure the accuracy of assigning pseudo-labels, we use a variety of distance indicators to supervise the distribution process of pseudo-labels: for the sample feature information of each dimension, we calculate the Euclidean distance, Manhattan distance, and Chebyshev distance from each test sample to each training sample, to obtain multi-distance indicators. By sorting these distance indicators, we select the nearest m unlabeled samples of each training sample to assign the same class of pseudo-labels, and the samples with statistical times not less than t are selected as effective pseudo-label samples to complete semi-supervised ensemble learning training.

The pseudo-code corresponding to the above process is illustrated in Algorithm 1. In the second step, we generate distance vectors based on multiple distance indicators and sort them in ascending order to match the nearest m unlabeled samples for each labeled sample. In the third step, we count the selected times of these unlabeled samples in each camera class and mark the unlabeled samples not less than the statistical time threshold t as valid pseudo-label samples, to realize a round of pseudo-label sample filtering.

Among them, to obtain more a priori information as much as possible in the case of few-shot samples, we use LBP and CFA image statistical features. The relevant details will be described in Section 4.

In addition, semi-supervised ensemble learning based on three distance indicators is the focus of our work. In this process, we have carried out two rounds of screening to ensure that the selected pseudo-label samples are as authentic and reliable as possible. Firstly, we get enough pseudo-label samples through the distance threshold m , and then, we select sufficiently reliable pseudo-label samples through the statistics threshold t . After two thresholds, we can ensure that the final pseudo-label samples are close enough to labeled samples in the spatial domain, so that the distribution of the pseudo-labels can be reliable. Therefore, by calculating multiple distances separately, we can combine the characteristics of multiple distance indicators and use the idea of ensemble learning to select more reliable pseudo-label samples.

After obtaining valid pseudo-label samples, we use SVM self-correction to correct some labels. Through the SVM classifier trained based on few-shot samples and selected pseudo-label samples to retest all unlabeled samples, we can get new pseudo-label samples and then repeat this process until the model converges, to complete the process of SVM self-correction. Finally, we can get the final classification accuracy on the testing data sets, as shown in Figure 3. We

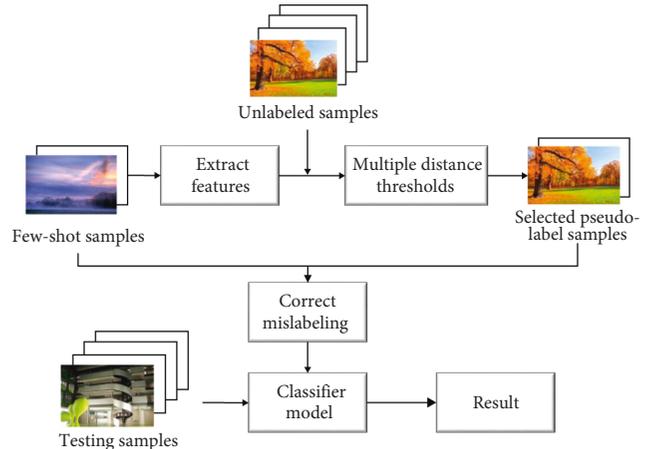


FIGURE 1: Framework of multi-DS algorithm.

use labeled samples to guide the selected pseudo-label samples for semi-supervised learning. At the same time, we update iteratively based on the results of each round of SVM classifier and then train the known labeled samples and the updated pseudo-label samples with SVM classifier to get the final model. The details of relevant strategy will be described in Section 4.

In this process, the reliable pseudo-label samples are the samples with high reliability screened through multiple thresholds, but they may not correspond to the real labels. At the same time, the pseudo-label samples also expand the training samples, making the available information more rich and reliable.

With these reliable pseudo-label samples, in the model training stage, we can guide and supervise the labeled samples with the help of the information of the pseudo-label samples. In this way, we can effectively enrich the information of the few-shot sample data sets and finally complete the process of semi-supervised learning.

4. Algorithm-Related Details

4.1. Feature Extraction Algorithm. To verify the rationality of our proposed multi-DS strategy, we combine the two characteristics of LBP and CFA. These two features have different generalization performances for image features. Among them, the LBP feature with equivalent mode effectively reduces the noise impact of high-frequency mode and has a strong ability to generalize features, while the CFA feature shows the inherent algorithmic differences in the cameras. The inherent difference between camera models is strong, so these two features can extract the inherent information of the image from two different aspects, to realize the extraction of camera features more scientifically.

4.1.1. CFA Features. CFA interpolation algorithm is a widely used model-based variable camera feature. Swaminathan et al. [21] calculated the pixel interpolation of each color channel, respectively, and calculated the CFA interpolation coefficient through the linear model. Taking the green

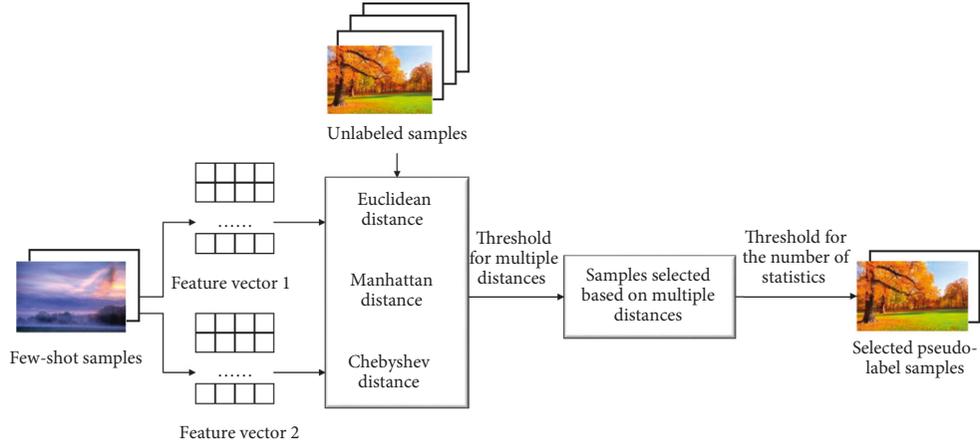


FIGURE 2: Selecting pseudo-label samples based on multiple distances.

Symbols: \mathcal{U} : Set of unlabeled samples \mathcal{L} : Set of labels \mathcal{S}_l : Set of samples with label $l \in \mathcal{L}$ \mathcal{M} : Collection of distance measures m : The distance threshold t : The statistics threshold

(1) Extract multiple feature vectors from all training samples;

(2) Calculate multiple distance parameters as follows:

for $i \in \mathcal{M}$ and $s \in \{\mathcal{S}_l: l \in \mathcal{L}\}$ **do**Form distance vector $D_{s,i} = \{d_{s,u}^{(i)}: u \in \mathcal{U}\}$ Sort $D_{s,i}$ in ascend order and take the first m entries to form a new vector $D'_{s,i}$ **end for**

(3) Selecting pseudo-label samples:

for $u \in \mathcal{U}$ **do****for** $l \in \mathcal{L}$ **do**Count for u in $\{D'_{s,i}: s \in \mathcal{S}_l, i \in \mathcal{M}\}$ **if** Total count of u is greater than a selected threshold t **then**Label u with pseudo-label l (Note: one unlabeled u may be marked with multiple pseudo-labels and appended to corresponding labeled sets)**end if****end for****end for**

ALGORITHM 1: Selecting pseudo-label samples based on multiple distances.

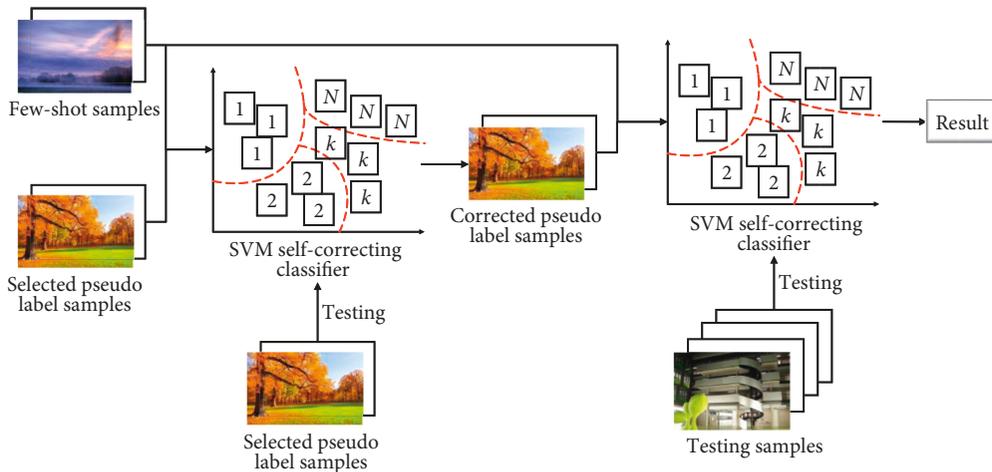


FIGURE 3: SVM self-correction model based on semi-supervised training.

channel of the image as an example, assuming that the position coordinate of the selected interpolation point is (m, n) and the selected neighborhood size is the pixel area of $(2k + 1) \times (2k + 1)$, the interpolation model of N interpolation pixels is shown as follows:

$$\begin{pmatrix} G_1 \\ G_2 \\ \vdots \\ G_N \end{pmatrix} = \begin{pmatrix} a_1^g g_1^1 + \cdots + a_{(2k+1)^2-1}^g g_{(2k+1)^2-1}^1 \\ a_1^g g_1^2 + \cdots + a_{(2k+1)^2-1}^g g_{(2k+1)^2-1}^2 \\ \vdots \\ a_1^g g_1^N + \cdots + a_{(2k+1)^2-1}^g g_{(2k+1)^2-1}^N \end{pmatrix} + \begin{pmatrix} a_1^r r_1^1 + \cdots + a_{(2k+1)^2}^r r_{(2k+1)^2}^1 \\ a_1^r r_1^2 + \cdots + a_{(2k+1)^2}^r r_{(2k+1)^2}^2 \\ \vdots \\ a_1^r r_1^N + \cdots + a_{(2k+1)^2}^r r_{(2k+1)^2}^N \end{pmatrix} + \begin{pmatrix} a_1^b b_1^1 + \cdots + a_{(2k+1)^2}^b b_{(2k+1)^2}^1 \\ a_1^b b_1^2 + \cdots + a_{(2k+1)^2}^b b_{(2k+1)^2}^2 \\ \vdots \\ a_1^b b_1^N + \cdots + a_{(2k+1)^2}^b b_{(2k+1)^2}^N \end{pmatrix}. \quad (1)$$

Among them, a_i^g , a_i^r , and a_i^b are the CFA interpolation coefficient weights of the green, red, and blue channels in the color image, respectively. g_α^k is the α interpolation coefficient of the neighborhood of the k pixel in the green channel. Similarly, r_α^k and b_α^k represent the corresponding coefficients of the red channel and the blue channel, respectively. Equation (2) can be abbreviated into the vector form of the following equation:

$$G = a_1^g g_1 + \cdots + a_{(2k+1)^2-1}^g g_{(2k+1)^2-1} + a_1^r r_1 + \cdots + a_{(2k+1)^2}^r r_{(2k+1)^2} + a_1^b b_1 + \cdots + a_{(2k+1)^2}^b b_{(2k+1)^2}. \quad (2)$$

The interpolation coefficients of other red channels and blue channels of color images can be similarly calculated by this strategy. Finally, the interpolation coefficients on the three color channels are obtained. For a given digital image, if we set $k = 3$, the neighborhood size of the interpolation point is 7×7 . According to the 2×2 Bayer CFA structure, the G interpolation coefficients of R and B sampling points and the R and B interpolation coefficients of two G sampling points are estimated, respectively, and a total of $((2k + 1)^2 - 1) \times 5 = 240$ interpolation coefficients are obtained. The mean and variance of 240 dimensional CFA interpolation coefficients are integrated into 480 dimensional CFA features.

4.1.2. LBP Features. LBP feature is an operator that reflects the local texture features of an image. It has the advantages of gray invariance, rotation invariance, and so on. The original LBP operator takes the gray value of the pixel of the center point as the standard threshold in the window of 3×3 , and the gray value of the other eight points is 1 if it is greater than the threshold and 0 if it is less than the threshold. After

comparison, the eight bit binary number will be converted into decimal, which is the LBP value of the center point. The calculation formula is shown as follows:

$$\text{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p, \quad (3)$$

$$s(x) = \begin{cases} 1, & x \geq 0, \\ 0, & x < 0. \end{cases} \quad (4)$$

Among them, parameter g_c is the gray value of the central point pixel, g_p represents the gray value of the adjacent points around the central point, and P is the total number of the surrounding adjacent points around the central pixel with R as the radius, where $p = 1, 2, \dots, P$. $s(x)$ is the binarization threshold processing function.

Considering that there are $2^8 = 256$ modes of LBP features for 8 neighborhood pixels, Ojala et al. [22] proposed a uniform pattern for dimensionality reduction: since the LBP feature mode of most images only jumps between 0 and 1 twice, the researchers defined the ‘‘equivalent mode’’ as that the number of numerical jumps after one rotation is less than 3; on the contrary, other modes become ‘‘mixed mode’’ (nonuniform pattern); that is, the LBP features of the mixed mode are integrated and the interference of high-frequency noise is reduced. Accordingly, the dimension of the features is also reduced from 256 to 59. Then, the 59 dimensional LBP features are extracted from the spatial domain, prediction error domain, and wavelet transform domain of the image, respectively. Color images have three color channels, and the post-processing algorithms of red channel and blue channel images are basically the same, so only LBP features of red channel and green channel need to be extracted. A total of $59 \times 3 \times 2 = 354$ dimensional LBP features are obtained.

4.2. Multiple Distance Indicators. We know that the essence of norm is distance, and its significance is to achieve comparison, so we can use a variety of distance indicators to measure distance. Here, we use one norm, two norms, and infinite norms for comprehensive judgment. They are also called Euclidean distance (ED), Manhattan distance (MD), and Chebyshev distance (CD).

The three distances are the measurement parameters for the characteristics of image samples, and their expressions in two-dimensional space are in the following formulas:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}, \quad (5)$$

$$d = |x_1 - x_2| + |y_1 - y_2|, \quad (6)$$

$$d = \max(|x_2 - x_1|, |y_2 - y_1|). \quad (7)$$

4.3. Semi-Supervised Learning. Semi-supervised learning (SSL) mainly aims at the problem of algorithm failure in the case of few-shot data. The characteristic of this strategy is

that it does not introduce external information and introduces unlabeled samples with the help of few-shot sample labels and some criteria conditions, to make the greatest use of the information of unlabeled samples from various ways. The expansion of finite sample set is realized to better train the model.

4.4. The Idea of Ensemble Learning. The idea of ensemble learning is to combine multiple weak supervision models to achieve a more comprehensive strong supervision model. The common combination methods are average method, voting method, and learning method.

By combining the different characteristics of multiple distance indicators, we use the idea of ensemble learning to select the pseudo-label samples closer to the labeled samples in the spatial domain. Because of the flexibility of the multiple distance indicators, it is possible to improve the reliability of the selected pseudo-label samples by integrating multiple distance indicators.

4.5. The SVM Self-Correction Classifier. The goal of the SVM classifier is to find a hyperplane to separate two classes of data, making the maximum margin between the samples. Multi-class classification can be realized by reusing SVM classifier. Among them, hyperplane refers to the separation surface that can divide the n-dimensional space into two parts. Support vector is the sample point closest to the classification hyperplane, and the geometric interval is usually used as the distance measure:

$$\gamma^{(i)} = y^{(i)} \left(\left(\frac{\omega}{\|\omega\|} \right)^T x^{(i)} + \frac{b}{\|\omega\|} \right). \quad (8)$$

Among them, y represents the class label of the data point, and ω and b are the parameters of hyperplane $\omega^T x + b = 0$. By continuous iterative training, the optimal hyperplane with the largest edge can be found. This iterative process is the training process of SVM classifier.

In our strategy, SVM classifier is used to classify the feature vectors of camera photographs. After identifying the selected pseudo-label samples, we put few-shot labeled and selected pseudo-label samples with labels and multiple feature vectors into the SVM for training. Then, we test with these pseudo-label samples, thereby continually iterating and updating the labels of pseudo-label samples to complete the SVM self-correction training process.

5. Experimental Results and Analysis

5.1. Experimental Image Data Sets and Settings. To fully verify the effectiveness of distance based on semi-supervised ensemble learning, all experiments in this study use the public databases in the field of image forensics: Dresden database, VISION database, and SOCRatES database, which are some popular image data sets used in forensic research.

In this experiment, we select 16 different classes of equipment in the Dresden database [23], 10 different classes of equipment in the VISION database [24], and 10 different types

TABLE 1: Data set in experiments (Dresden).

Camera model	Abbr.
Canon_IXUS 70	C1
Casio_EX-Z150	C2
FujiFilm_FinePix J50	F1
Kodak_M1063	K1
Nikon_CoolPix S710	N1
Nikon_D70	N2
Nikon_D200	N3
Olympus_Mju_1050SW	O1
Panasonic_DMC-FZ50	P1
Praktica_DCZ5.9	P2
Rollei_RCP-7325XS	R1
Samsung_L74Wide	SL1
Samsung_NV15	SN1
Sony_DSC-H50	SD1
Sony_DSC-T77	SD2
Sony_DSC-W170	SD3

TABLE 2: Data set in experiments (VISION).

Camera model	Abbr.
Samsung_Galaxy S3	Sa1
Apple_iPhone 4s	Ap1
Huawei_P9	Hu1
LG_D290	Lg1
Apple_iPhone 5c	Ap2
Apple_iPhone 6	Ap3
Lenovo_P70A	Le1
Samsung_Galaxy Tab 3	Sa2
Apple_iPhone 4	Ap4
Microsoft_Lumia 640 LTE	Mi1

TABLE 3: Data set in experiments (SOC rates).

Camera model	Abbr.
Apple iPhone 5s	A1
Apple iPhone 6	A2
Apple iPhone 6s	A3
Apple iPhone 7	A4
Asus Zenfone 2	As1
LG G3	L1
Motorola Moto G	M1
Samsung Galaxy A3	SG1
Samsung Galaxy S5	SG2
Samsung Galaxy S7 Edge	SG3

of equipment in the SOCRatES database [25]. The specific equipment information is shown in Tables 1–3. Among them, the number of training samples is limited, ranging from 5, 10, 15, 20, and 25 images of each class. The test set consists of 150 unlabeled samples for each class, which means 2400 (in the Dresden data set) and 1500 (in the VISION and SOCRatES data sets) image samples in total. Based on our dynamic threshold selection strategy, different numbers of pseudo-label samples (from 25 to 55) were selected for each class, from these unlabeled samples. We expanded the training set using the pseudo-label samples as training samples. The final result is a stable statistical result obtained by averaging the results after 10 random training tests.

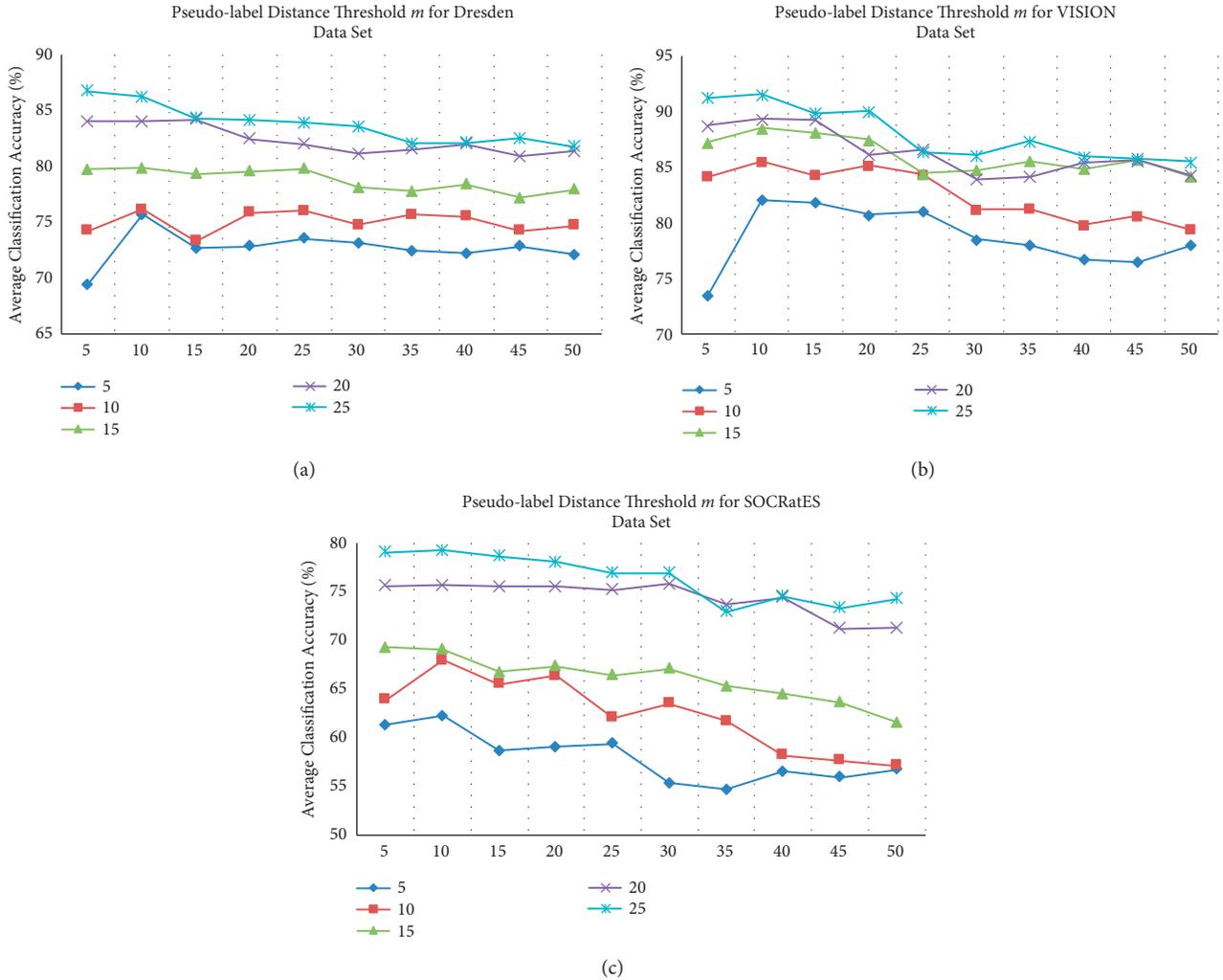


FIGURE 4: Accuracy versus the pseudo-label distance threshold m on multiple data sets, with respect to the size of labeled sample set ($n = 5, 10, 15, 20, 25$) for each label. (a) Dresden. (b) VISION. (c) SOCRatES.

5.2. Algorithm Performance Evaluation and Analysis. In this study, the experiment with LBP features is labeled as the LBP-SVM method, and the experiment with CFA features is labeled as the CFA-SVM method. The experiment with double features is labeled as the multi-SVM method. At the same time, for the experiment with the DS method using only one kind of features, the method using only LBP features is called LBP-DS, the method using only CFA features is called CFA-DS, and the final method integrating the two features is called multi-DS.

In this experiment, the number of training samples is 5, 10, 15, 20, and 25, respectively, which is to simulate the situation of different numbers of training samples, and there are two thresholds in this strategy. We conducted detailed experiments on each data set.

For the pseudo-label distance threshold of m , we tested the accuracy when the number of training samples was 5, 10, 15, 20, and 25, respectively. The pseudo-label distance threshold m ranges from 5 to 50, representing the number of unlabeled samples of each class selected based on multi-distance indicators. By testing the pseudo-label distance

threshold m with different sample numbers, we finally determined that the parameter m is 10, as shown in Figure 4.

For the statistical number threshold t , the experimental results are shown in Tables 4–6. When the number of training samples of each class is 5, 10, 15, 20, and 25, respectively, we test the accuracy of the value of different statistical time threshold t . For each labeled sample, we selected the nearest m unlabeled samples according to three distance thresholds. According to the number of times these unlabeled samples were selected, we set threshold t for each camera class, indicating the minimum number of times each unlabeled sample needs to be selected to be marked as a valid pseudo-labeled sample, that is, the statistical time threshold t . Finally, for different numbers of training samples, the threshold t of statistical times we choose is 3, 5, 6, 7, and 8, respectively. The reason why the threshold is selected in this way is that when the threshold is increased again, unlabeled samples will not be selected for some classes. When the threshold is reduced, the selected samples will be increased, resulting in a decrease in the purity of the selected sample set.

TABLE 4: Effect of statistical number threshold t on accuracy (%) (Dresden data set).

Threshold t	5	10	15	20	25
1	71.21	75.42	78.96	80.38	80.54
2	72.33	77.08	79.96	83.25	84.96
3	72.50	76.79	80.02	83.33	84.21
4	55.75	76.08	80.17	82.96	83.21
5	44.54	76.63	79.25	83.63	82.38
6	39.79	74.00	79.54	83.00	85.50
7	45.42	64.04	78.50	84.29	85.46
8	41.00	71.58	75.71	81.79	86.04

TABLE 5: Effect of statistical number threshold t on accuracy (%) (VISION data set).

Threshold t	5	10	15	20	25
1	77.60	81.47	88.27	88.73	89.07
2	82.13	83.93	88.00	88.73	90.87
3	82.27	83.20	87.07	89.00	90.73
4	75.27	79.87	87.87	87.60	91.20
5	66.47	85.40	88.13	89.47	91.40
6	70.07	81.27	89.80	89.13	90.47
7	71.67	79.40	85.67	89.67	91.27
8	57.60	67.93	85.33	88.60	91.53

TABLE 6: Effect of statistical number threshold t on accuracy (%) (SOCRAES data set).

Threshold t	5	10	15	20	25
1	52.53	66.00	70.53	76.07	75.53
2	53.00	67.27	70.27	74.87	77.87
3	52.53	67.80	69.87	76.87	78.33
4	52.60	67.07	66.73	72.80	76.87
5	52.33	68.13	68.53	74.60	76.93
6	48.13	64.60	70.73	74.87	78.73
7	47.73	64.73	70.20	76.27	79.20
8	47.07	62.33	69.20	76.80	80.40

In this study, 10 random experiments were carried out on several databases, and the results were averaged. The final experimental results in the three experimental databases are shown in Figures 5–7, and the confusion matrices between different classes are shown in Tables 7–9.

As shown in Figure 5, in the experiment of Dresden data set, when the number of training samples is 5, the accuracy of CFA-DS method is 18.6% higher than that of the CFA-SVM method, which fully reflects that the CFA-DS method has much better performance than the CFA-SVM method in the case of few-shot samples. However, in terms of LBP features, the performance of LBP-DS method is not satisfactory. Therefore, this study considers using the idea of ensemble learning to combine LBP features and CFA features, that is, multi-DS method, and carries out subsequent experiments on other data sets.

Table 7 shows the confusion matrix obtained by repeating the experiment 10 times when the number of training samples of multi-DS method is 25. Where the accuracy is less than 0.1%, it is marked as “—.” Through the

confusion matrix, we can see that Sony_DSC-H50 (SD1) model and Sony_DSC-W170 (SD3) model are 66.5% and 47.4%, respectively, and the degree of mutual confusion is the greatest. Through consulting the data, we find that the results are still the same in many previous research works, which are the same as that in the strategy proposed in this study. This is because the cameras of the two models adopt similar image post-processing algorithms. The difference between camera models is very low.

From the results of multiple experimental data sets, the performance improvement is more obvious when the number of training samples is less. The three experimental databases show similar results, which shows that the strategy proposed in this study is a good solution to the problem of few-shot samples. Therefore, the experimental results of this study can fully reflect the wide applicability and universality of the strategy proposed in this study and help to solve the actual judicial evidence problem of insufficient known samples.

At the same time, to validate the stability of the model, we conduct a set of stability experiments. When the number

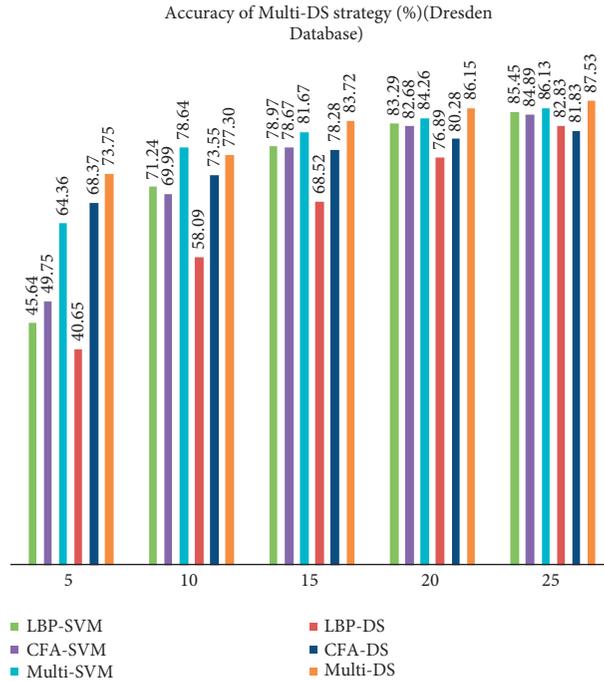


FIGURE 5: Results of multi-DS strategy (%) (Dresden database).

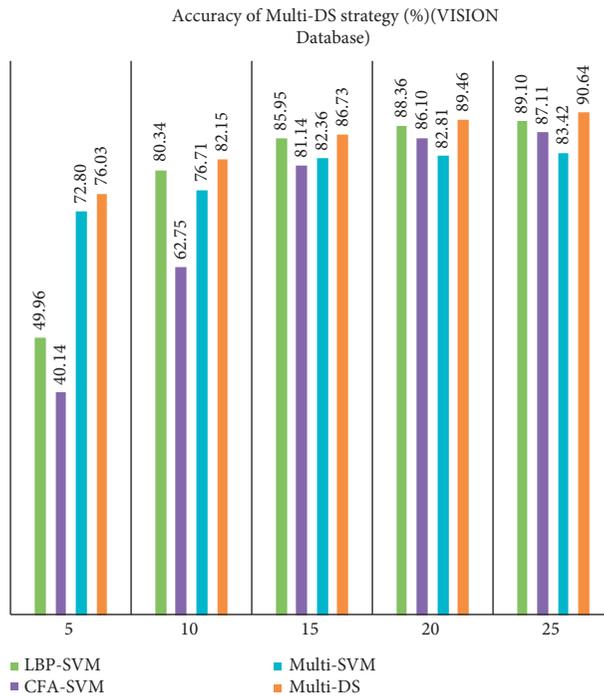


FIGURE 6: Results of multi-DS strategy (%) (VISION database).

of samples in each class is quite small, randomly selected samples may not necessarily represent the whole sample population. Therefore, when the number of few-shot samples in each class is only 5, we conduct 20 experiments and average the results as shown in Table 10. We thicken the maximum and minimum values in each data set. The experimental results show that our model is not stable with few-shot samples.

In addition, we compare the proposed multi-DS strategy with other existing methods to verify the performance of our strategy. We reselected 14 (in the Dresden data set), 11 (in the VISION data set), and 10 (in the SOCRATES data set) camera models and 10 labeled samples of each class. These reselected camera classes have the same class settings as the data set in the paper [19] to ensure fairness in our performance evaluation. The experimental

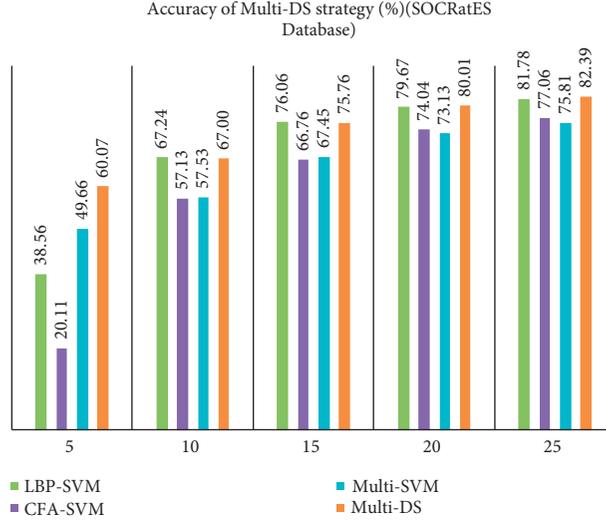


FIGURE 7: Results of multi-DS strategy (%) (SOCRateS database).

TABLE 7: Average confusion matrix obtained by SVM classification over 20 iterations (%) (Dresden database).

	C1	C2	F1	K1	N1	N2	N3	O1	P1	P2	R1	SL1	SN1	SD1	SD2	SD3
C1	97.5	0.7	0.2	—	0.2	0.1	—	0.3	0.1	0.3	0.1	0.5	—	—	—	—
C2	2.2	90.9	0.1	0.1	0.7	—	—	—	0.6	0.8	—	3.6	—	0.3	0.5	0.2
F1	0.1	—	86.3	0.1	—	0.4	0.3	0.1	1.8	1.5	8.2	0.5	0.5	—	—	0.1
K1	—	—	0.3	94.6	—	0.7	3.6	—	—	0.7	—	—	0.1	—	—	—
N1	0.7	2.3	—	0.7	92.2	1.3	0.8	0.2	—	0.4	—	1.4	—	—	—	—
N2	0.5	0.7	—	1.1	0.7	91.9	3.3	0.1	—	0.3	0.4	0.7	0.3	—	—	—
N3	—	—	0.5	0.8	—	2.9	95.0	—	0.1	1.0	0.2	—	—	0.1	0.3	—
O1	2.2	—	0.3	0.2	0.1	0.5	0.5	93.5	0.1	0.3	0.1	2.1	0.1	—	—	—
P1	0.1	0.9	0.1	0.1	—	0.1	2.0	0.1	93.7	0.2	0.1	2.2	0.2	—	0.1	0.1
P2	0.4	1.3	0.9	1.4	2.1	2.6	—	—	0.8	87.2	1.9	—	1.2	0.1	—	0.1
R1	—	0.1	3.7	0.1	0.5	2.2	0.1	0.1	—	1.0	90.4	0.4	1.4	—	—	—
SL1	1.0	2.1	0.1	—	0.9	1.3	0.2	0.3	0.8	0.4	0.5	92.0	0.4	—	—	—
SN1	—	0.9	1.4	0.8	0.1	0.6	0.1	0.1	0.7	4.3	2.4	0.7	87.4	0.5	—	—
SD1	0.1	0.1	0.3	0.1	—	—	0.9	—	0.2	0.3	0.4	—	0.2	66.5	3.7	27.2
SD2	—	—	0.2	—	—	—	0.7	—	—	0.1	0.4	0.2	—	2.0	93.9	2.5
SD3	—	0.1	0.1	—	—	—	0.7	—	—	0.1	0.1	—	—	50.2	1.3	47.4

The bold values (diagonal values) indicate that the prediction labels of the model are equal to the real labels, that is, the accuracy of the correctly classified results.

TABLE 8: Confusion matrix of the results of multi-DS (%) (VISION database).

	Sa1	Ap1	Hu1	Lg1	Ap2	Ap3	Le1	Sa2	Ap4	Mi1
Sa1	93.5	0.1	2.5	0.9	0.4	1.6	0.8	0.3	—	—
Ap1	—	76.2	1.9	1.2	12.0	4.0	2.6	0.9	—	1.1
Hu1	1.2	2.6	87.7	2.9	0.2	2.1	3.2	—	—	—
Lg1	0.1	0.3	2.6	90.9	1.0	1.3	3.6	—	—	0.1
Ap2	0.1	8.9	0.7	1.2	84.0	3.0	0.7	0.6	0.1	0.7
Ap3	0.6	2.9	2.1	1.8	3.9	86.3	1.7	0.7	—	—
Le1	0.1	0.2	1.5	1.9	0.7	0.2	95.1	—	—	0.4
Sa2	0.9	—	0.8	0.1	—	—	—	98.2	—	—
Ap4	—	0.1	—	—	0.4	0.2	—	—	98.9	0.4
Mi1	0.1	0.5	1.0	1.4	0.2	0.5	0.5	—	0.3	95.6

The bold values (diagonal values) indicate that the prediction labels of the model are equal to the real labels, that is, the accuracy of the correctly classified results.

results between us and other existing methods are shown in Table 11. In the same experimental setup, our experimental results have better performance than other existing methods.

In addition, we added the experiment of the influence of the number of classes on the classification accuracy. The independent variables range from only part of the data set at the model level to the complete data set. The number of

TABLE 9: Confusion matrix of the results of multi-DS (%) (SOCRatES database).

	A1	A2	A3	A4	As1	L1	M1	SG1	SG2	SG3
A1	81.6	6.4	5.4	2.8	0.5	0.2	0.4	0.7	1.6	0.3
A2	4.6	71.4	6.5	5.0	0.9	1.0	2.7	2.8	2.5	2.6
A3	3.4	7.1	78.8	5.2	1.8	0.1	0.4	0.6	0.8	1.8
A4	3.3	3.9	9.1	76.8	2.1	0.5	1.5	0.4	0.9	1.4
As1	1.0	0.5	2.8	3.8	85.6	1.9	1.2	1.1	1.4	0.8
L1	0.4	0.9	0.3	1.3	2.8	88.5	1.0	0.7	3.4	0.6
M1	0.5	2.5	1.0	5.1	3.2	0.9	81.9	1.2	2.2	1.5
SG1	2.3	1.4	0.3	0.3	0.5	0.1	0.8	88.0	2.6	3.6
SG2	3.0	4.2	0.8	2.0	2.1	6.3	1.5	2.8	86.0	1.3
SG3	0.2	1.6	1.7	1.8	2.8	1.5	1.6	2.6	1.0	85.2

The bold values (diagonal values) indicate that the prediction labels of the model are equal to the real labels, that is, the accuracy of the correctly classified results.

TABLE 10: Stability analysis with 5 few-shot samples per class.

Results	Dresden	VISION	SOCRatES
1	78.83	76.53	59.80
2	78.64	78.60	58.80
3	72.13	71.00	62.27
4	75.79	72.33	60.33
5	72.96	83.07	61.47
6	72.88	77.40	61.93
7	69.58	74.87	62.33
8	64.29	80.00	59.67
9	78.42	69.20	65.60
10	75.21	77.13	56.67
11	76.54	80.07	57.67
12	79.21	77.40	57.73
13	69.21	75.63	61.87
14	70.00	74.60	51.27
15	69.33	72.87	61.33
16	77.42	78.27	61.13
17	77.21	76.00	62.60
18	73.25	74.07	57.27
19	76.25	73.73	63.40
20	67.83	77.73	58.33
Average	73.75	76.03	60.07

The bold values represent the maximum value, minimum value, and average value of each column to indicate the variation range.

TABLE 11: Camera identification accuracy compared with the previous methods (%).

Method	Dresden	VISION	SOCRatES
EP [17]	73.84	79.94	63.91
MTDEM [26]	75.16	80.49	64.84
Deep Siamese Network [19]	85.30	75.20	—
Multi-DS	86.08	85.56	67.00

The bold values indicates the maximum value of each column to indicate the best performing approach on each dataset.

camera model classes in the Dresden database ranges from 14 to 27 (the whole data set), while the number of classes in VISION database ranges from 11 to 35 (the whole data set). The experimental results show that the classification

accuracy will gradually decrease when the number of models increases. We verified this result on the Dresden and VISION databases in Figures 8 and 9, which is consistent with the results in [19].

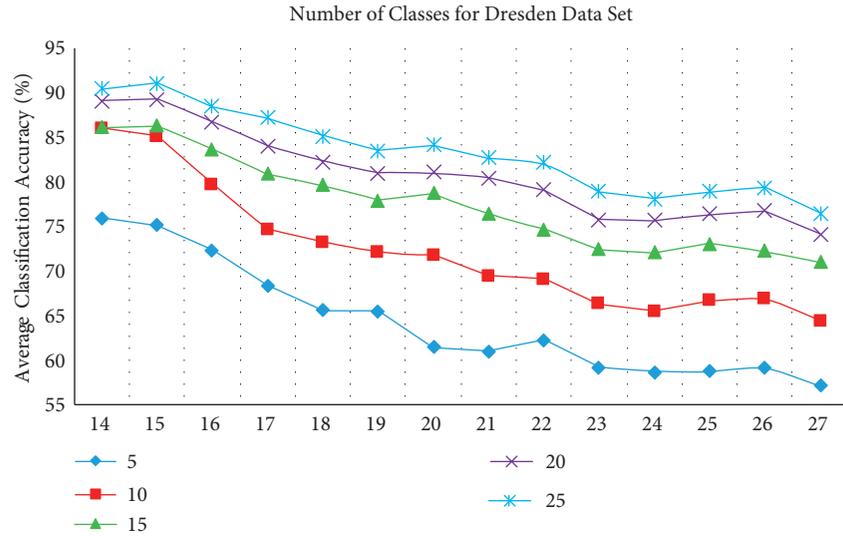


FIGURE 8: Impact of number of classes on accuracy (Dresden database).

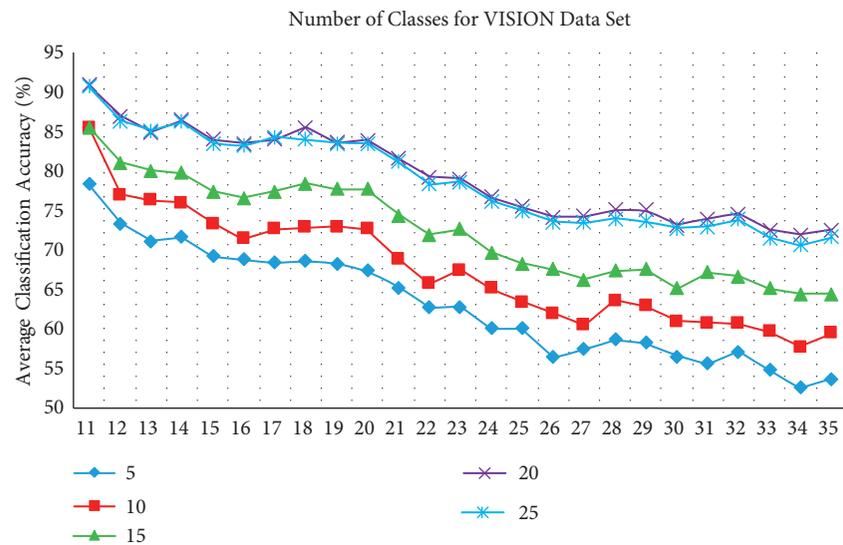


FIGURE 9: Impact of number of classes on accuracy (VISION database).

6. Conclusion

In this study, we solve the problem of few-shot label classification in source camera recognition. Due to the practical significance of the few-shot label scenario, it is very important to deal with this challenge effectively. Therefore, we propose a (multi-DS) strategy to solve this problem. Through the ensemble learning of various types of distance indicators and the self-correction of SVM to obtain the unlabeled sample information, the available information is effectively supplemented. Our experimental results show that this strategy can effectively improve the image traceability accuracy when facing few-shot sample data sets and provide a good solution for the actual judicial forensic problem with insufficient number of known samples. In the future work, the actual

judicial forensic task of source camera identification with fewer shot samples or in extreme cases deserves our attention and discussion.

Data Availability

The data sets used or analyzed in this study are published data sets: the Dresden data set is unavailable at present, possibly be restricted to protect the patient privacy. The Dresden data used to support the findings of this study are available from the corresponding author upon request. The data set is derived from this paper: Gloe, Thomas, and Rainer Böhme. "The 'Dresden Image Database' for benchmarking digital image forensics." Proceedings of the 2010 ACM symposium on applied computing. 2010:1584–1590. The VISION data set is

available from this link: <https://lesc.dinfo.unifi.it/en/datasets/> The SOCRatES data set is available from this link: <http://socrates.eurecom.fr/>

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. U1936117, 62106037, and 62076052), the Science and Technology Innovation Foundation of Dalian (No. 2021JJ12GX018), the Open Project Program of the National Laboratory of Pattern Recognition (NLPR) (No. 202100032), and the Fundamental Research Funds for the Central Universities (No. DUT21GF303).

References

- [1] A. T. Ho and S. Li, *Handbook of Digital Forensics of Multimedia Data and Devices*, John Wiley & Sons, Hoboken, NJ, USA, 2015.
- [2] J. Luka, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.
- [3] X. Hu, Z. Yang, G. Liu, Q. Liu, and H. Wang, "Virtual label expansion-highlighted key features for few-shot learning," in *Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, Shenzhen, China, July 2021.
- [4] B. Wang, S. Wu, F. Wei, Y. Wang, J. Hou, and X. Sui, "Virtual sample generation for few-shot source camera identification," *Journal of Information Security and Applications*, vol. 66, Article ID 103153, 2022.
- [5] T. Vu, M.-T. Luong, Q. Le, G. Simon, and M. Iyyer, "Strata: self-training with task augmentation for better few-shot learning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5715–5731, Punta Cana, Caribbean, November 2021.
- [6] J. Zhou, Y. Zheng, J. Tang, L. Jian, and Z. Yang, "Flipda: effective and robust data augmentation for few-shot learning," *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 8646–8665, 2022.
- [7] U. Osahor and N. M. Nasrabadi, "Ortho-shot: low displacement rank regularization with data augmentation for few-shot learning," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2200–2209, Waikoloa, HI, USA, January 2022.
- [8] K. Huang, J. Geng, W. Jiang, X. Deng, and Z. Xu, "Pseudo-loss confidence metric for semi-supervised few-shot learning," in *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8651–8660, Montreal, Canada, October 2021.
- [9] K. Ding, J. Wang, J. Caverlee, and H. Liu, "Meta Propagation Networks for Graph Few-Shot Semi-supervised Learning," 2022, <https://arxiv.org/abs/2112.09810>.
- [10] B. Zhang, H. Ye, G. Yu et al., "Sample-centric feature generation for semi-supervised few-shot learning," *IEEE Transactions on Image Processing*, vol. 31, pp. 2309–2320, 2022.
- [11] M. Kharrazi, H. T. Sencar, and N. Memon, "Blind source camera identification," in *Proceedings of the 2004 International Conference on Image Processing ICIP'04*, pp. 709–712, IEEE, Singapore, April 2004.
- [12] S. Lyu and H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines," in *Proceedings of the International Workshop on Information Hiding*, pp. 340–354, Springer, Noordwijkerhout, The Netherlands, October 2002.
- [13] I. Avcibas, N. Memon, and B. Sankur, "Steganalysis using image quality metrics," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 221–229, 2003.
- [14] O. Çeliktutan, B. Sankur, and I. Avcibas, "Blind identification of source cell-phone model," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 553–566, 2008.
- [15] A. Swaminathan, M. Wu, and K. R. Liu, "Component forensics of digital cameras: a non-intrusive approach," in *Proceedings of the 2006 40th Annual Conference on Information Sciences and Systems*, pp. 1194–1199, IEEE, Princeton, NJ, USA, March 2006.
- [16] B. Xu, X. Wang, X. Zhou, J. Xi, and S. Wang, "Source camera identification from image texture features," *Neurocomputing*, vol. 207, pp. 131–140, 2016.
- [17] Y. Tan, B. Wang, M. Li, Y. Guo, X. Kong, and Y. Shi, "Camera source identification with limited labeled training set," in *Proceedings of the 2015 International Workshop on Digital Watermarking*, pp. 18–27, Springer, Tokyo, Japan, October 2015.
- [18] X. Liang, Y. Zhang, and J. Zhang, "Attention multisource fusion-based deep few-shot learning for hyperspectral image classification," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 8773–8788, 2021.
- [19] V. U. Sameer and R. Naskar, "Deep siamese network for limited labels classification in source camera identification," *Multimedia Tools and Applications*, vol. 79, no. 37–38, pp. 28079–28104, 2020.
- [20] Y. Huo, J. Guan, J. Zhang, M. Zhang, J.-R. Wen, and Z. Lu, "Zero-shot learning with few seen class samples," in *Proceedings of the 2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1336–1341, IEEE, Shanghai, China, July 2019.
- [21] A. Swaminathan, M. Wu, and K. R. Liu, "Nonintrusive component forensics of visual sensors using output images," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 1, pp. 91–106, 2007.
- [22] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.
- [23] T. Gloe and R. Böhme, "The'dresden image database'for benchmarking digital image forensics," in *Proceedings of the 2010 ACM Symposium on Applied Computing*, pp. 1584–1590, Sierre, Switzerland, March 2010.
- [24] D. Shullani, M. Fontani, M. Iuliani, O. A. Shaya, and A. Piva, "Vision: a video and image dataset for source identification,"

- EURASIP Journal on Information Security*, vol. 2017, no. 15, pp. 1–16, 2017.
- [25] C. Galdi, F. Hartung, and J.-L. Dugelay, “A Database of Realistic Data for Source Camera Recognition on Smartphones,” in *Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods ICPRAM*, pp. 648–655, Prague, Czech Republic, January 2019.
- [26] S. Wu, B. Wang, J. Zhao, M. Zhao, K. Zhong, and Y. Guo, “Virtual sample generation and ensemble learning based image source identification with small training samples,” *International Journal of Digital Crime and Forensics*, vol. 13, no. 3, pp. 34–46, 2021.