

# Blind Identification of Source Cell-Phone Model

Oya Çeliktutan, Bülent Sankur, *Senior Member, IEEE*, and İsmail Avcıbaşı, *Member, IEEE*

**Abstract**—The various image-processing stages in a digital camera pipeline leave telltale footprints, which can be exploited as forensic signatures. These footprints consist of pixel defects, of unevenness of the responses in the charge-coupled device sensor, black current noise, and may originate from proprietary interpolation algorithms involved in color filter array. Various imaging device (camera, scanner, etc.) identification methods are based on the analysis of these artifacts. In this paper, we set to explore three sets of forensic features, namely binary similarity measures, image-quality measures, and higher order wavelet statistics in conjunction with SVM classifiers to identify the originating camera. We demonstrate that our camera model identification algorithm achieves more accurate identification, and that it can be made robust to a host of image manipulations. The algorithm has the potential to discriminate camera units within the same model.

**Index Terms**—Binary similarity measures (BSM), feature selection, image forensics, interpolation (demosaiçing) algorithm.

## I. INTRODUCTION

WE ADDRESS the problem of identification of the source camera of images. More specifically, we intend to develop a classification algorithm to determine the model and brand of the camera with which an image was acquired. Without loss of generality, the experiments are conducted on cell-phone cameras. First because they are much more ubiquitous, and also because they are becoming technologically on a par with commercial electronic cameras. In fact, according to GfK Group's "2007 Mid-Year Pan Asian Mobile Phone and Digital Imaging" report,<sup>1</sup> 78 million camera phones were sold in the first half of 2007 in Asia and in the near future there will be several hundred million of people using them. In the forensic context, the wide-scale availability of cellular-phone cameras will signify that there will be increasingly more evidence in the form of cell-phone images brought to courts or reported to law-enforcement officers. The downside of it is that cell-phone cameras can also be used for criminal purposes, such as pilfering credit-card information and child pornography. Therefore, the identification and/or verification of source cameras has become necessary for legal purposes and security investigations.

Manuscript received November 15, 2007; revised May 13, 2008. First published July 25, 2008; last published August 13, 2008 (projected). This work was supported by TUBITAK under Projects 104E056 and 107E001. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hany Farid.

O. Çeliktutan and B. Sankur are with the Electrical-Electronics Engineering Department, Boğaziçi University, İstanbul 34342, Turkey (e-mail: oya.celiktutan@boun.edu.tr; bulent.sankur@boun.edu.tr).

İ. Avcıbaşı is with the Electrical-Electronics Engineering Department, Başkent University, Ankara 06530, Turkey (e-mail: avcibas@baskent.edu.tr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>

Digital Object Identifier 10.1109/TIFS.2008.926993

<sup>1</sup>[Online]. Available: <http://www.cameraphonereport.com/>

The camera identification problem can be addressed within the framework of image forensics. Image forensics is an emerging field concerned with determining the source and potential authenticity of digital objects and possibly reconstructing the history of manipulations effected. An obvious threat to image authenticity is the ease with which digital images can be created, edited, and manipulated with sophisticated tools which do not leave much perceptible trace [25]. A second important threat is the obfuscation of the source identity of the imaging device. In this respect, the image header, where such information as camera brand, model, date, and time, etc., are embedded is no longer reliable. Digital watermarking falls short of meeting all desiderata of these two problems. Forensic tools, however, can be designed to recognize the nature and location of the manipulation and to identify the source device of the image.

In this study, we develop a classification algorithm for source cameras to identify the camera brand and model from any single given sample image. We use the term "camera brand" to denote the manufacturer (e.g., Nokia or Siemens), and we use the term "model" to denote alternate products from the same manufacturer (e.g., Nokia 5140, 6230, or 6600). The pinpointing of the imaging sensor type is based on the telltale effects created within the proprietary image formation pipeline in cameras. These are artifacts due to color filter-array (CFA) interpolation as well as charge-coupled device (CCD) sensor defects and nonuniformities. In addition, the algorithm is intended to be robust against various manipulations, such as JPEG compression, conducted on the images. Our work differs from its correlates in the area of camera identification [4], [14] in that we explore feature fusion and decision fusion schemes on a larger category of features. We also investigate sensitivity of the identification performance against various image manipulations, that is, when the training set of images differs from the test set due to signal processing and/or geometric operations on images. Finally, our scheme proves to be superior under various scenarios than its competitors as in Table X.

This paper is organized as follows. Previous work is reviewed in Section II. The source camera-identification algorithm, from feature extraction to the decision fusion stage, is given in Section III. The details of the experiments and their results are provided in Section IV. In Section V, we discuss future work and draw our conclusions.

## II. PREVIOUS WORK ON CAMERA FORENSICS

There have been a number of studies in the literature for camera identification based on the exploitation of residual artifacts and imperfections in the imaging pipeline. We can divide these approaches into two main groups according to the information source they use. The methods in the first group use sensor noise and artifacts in the CCD array. The second

group approaches camera identification via demosaicing artifacts taking place in the processing of raw sensor images.

Camera identification based on sensor noise: Geradts *et al.* [10] observed that large CCD arrays often contain a variety of manufacturing defects, such as hot point defects, dead pixels, pixel traps, and cluster defects, which, in total, amount to fixed pattern noise. In addition, camera electronics in the camera generate random dark current. They have observed that while dark current has limited potential in building a forensic signature, the fixed pattern noise of the CCD array is instrumental in constructing a unique pattern for each camera. Kurosawa *et al.* [16] and Lukas *et al.* [18] also turned their attention to the pattern noise of CCD arrays. Lukas *et al.* found that the systematic part of the noise does not change much from image to image, is relatively stable over camera life span and operating conditions, and consists of the fixed pattern noise plus photoresponse nonuniformity artifacts. While the fixed pattern noise can be corrected for by subtraction of a dark frame, the photoresponse nonuniformity noise (PRNU) caused by pixel nonuniformities is a more persistent feature. The PRNU can be reliably extracted by averaging the denoising residuals of several images. This signal pattern plays the role of a signature, that is, it acts like a spread-spectrum watermark unique to each camera.

Camera identification based on demosaicking artifacts: Commercial imaging devices use a single mosaic structured CFA rather than having separate filters for each color component. Camera models employ their proprietary interpolation algorithm in recreating the missing color values [1]. The grid interpolation process, in turn, leaves footprints, such as correlation patterns between contiguous bit planes. Kharazzi *et al.* [14] tried to capture the differences in CFA configuration and color-processing pipeline by a feature-based approach. They focused on features, such as mean value of RGB channels, correlations between color components, differences in neighborhood distribution, wavelet-domain statistics [19] and image-quality measures. Extensions of this work can be found in [4], [5], and [28]. Since the residuals of interpolation algorithms depend on the nature of the captured content, these authors fine-tuned their algorithm by separately treating the smooth and nonsmooth parts of the images. In another study, Long and Huang [17] used interpixel correlations originating from demosaicking. They defined a quadratic pixel correlation model and obtained a coefficient matrix for each color band based on this model. Swaminathan *et al.* [27] investigated the demosaicking artifacts using an analysis-by-synthesis method. They divided the image into three regions based on gradient features in a local neighborhood and then they estimated interpolation coefficients through singular value decomposition (SVD) for each region and each color band separately. Then, they reinterpolated the sampled CFA pattern and chose the one that minimizes the difference between the estimated final image and actual image produced by the camera.

Camera identification by alternative methods: In [8], the authors used intrinsic lens radial distortion for camera identification. The underlying idea is that most of the digital cameras are equipped with lenses having spherical surfaces and the degree of their inherent radial distortions varies from one manufacturer to another. However, the lens features have not proven to be robust,

since the distortion parameters are influenced by the focal length of the lens. Another method is the use of sensor dust characteristics of digital single-lens reflex (DSLR) cameras [9]. When the interchangeable lens is removed, dust particles are attracted to the sensor and they create a dust pattern in front of the imaging sensor. This dust pattern may be fairly stable on the sensor surface since most digital cameras do not offer a built-in solution for sensor dust removal.

Our approach encompasses two main paradigms. We extract a number of features from images based on the assumption that the processing pipeline specific to a manufacturer, and the camera noise and CCD array nonuniformity will leave telltale vestiges on the images. The various statistical moments of the image denoising residuals, described in the sequel, will capture these effects by enabling camera model and brand discrimination.

Table I summarizes the main properties of available camera-identification algorithms.

### III. SOURCE CAMERA-IDENTIFICATION ALGORITHM

The block diagram of the algorithm to identify the class of source camera is given in Fig. 1. Given a single test image, the algorithm starts with extracting its forensic features. Previous studies on steganalysis and forensics have revealed different categories of features. We have two options: 1) pool all of these disparate features (essentially feature fusion) and build one classifier and 2) treat the features in separate categories with apposite classifiers and fuse them at the decision level.

In the sequel, we describe the features used, the classification method, and the fusion procedure.

#### A. Forensic Feature Types for Cameras

We used three varieties of forensic features. They share the commonalities, first, of being as independent as possible from image content, and second, capturing noise-like residuals in the image due to the artifacts, due to each camera's specific processing pipeline. These features were described in detail in previous work [2]–[4], [19]; hence, we only give brief descriptive information here.

Characteristics of the lower order bit planes: Based on the conjecture that the camera artifacts will reside mostly in the less-significant bit planes, we considered the characteristics of the neighborhood bit patterns in the 6th, 7th, and 8th bit planes. The occurrence of bit patterns within  $3 \times 3$  neighborhoods was quantified via the method of local binary pattern (LBP) method [20], [21]. The histograms of these patterns were computed in three modalities, namely, spatial, quantal, and chromatic. More explicitly, spatial patterns are those that occur within a bit plane; quantal features are those that occur between two adjacent bit planes (say, between 7th and 8th bit planes), and finally, chromatic patterns are those that occur across color components (e.g., RGB bands). The organization of these patterns is illustrated in Fig. 2 and more details are given in Appendix A. Since we consider the similarity between histograms or moments of binary patterns we call them binary similarity measures (BSM). The selected combinations of these measurements have amounted to 480 BSM features [3], [4].

TABLE I  
COMPARISON OF METHODS FOR IMAGE-ACQUISITION DEVICE IDENTIFICATION

Method	Property	Comments
CFA interpolation [5]	Treats smooth and non-smooth parts separately, and uses interpolation coefficients and periodicity in the variance of the second order derivative of interpolated image as features in conjunction with a SVM classifier.	Affected by the JPEG compression in estimating the interpolation algorithm.
Sensor noise [7], [18]	Constructs a reference fixed noise pattern from averaging multiple noise residuals obtained by wavelet denoising filter, and determines the presence of noise pattern in a test image by using a correlation predictor.	Noise residual affected by the image content in busy parts, hence must be extracted by flat, non-saturated regions of the image; geometrical transformations, such as downsampling and cropping, deform the noise pattern.
Lens radial distortion [8]	It estimates radial distortion parameters of the lens, and uses SVM for classification.	Influenced by the focal length of the lens.
Sensor dust characteristics [9]	Uses dust model of the DSLR camera	Dust specks are hardly detected in non-smooth and complex regions of the image; moreover, they become nearly invisible for wide apertures.
CFA interpolation [27]	Constructs a search space of CFA patterns, estimate color interpolation coefficients via linear approximation and find the CFA pattern that produces minimum interpolation errors.	There are a limited number of interpolation algorithms and CFA patterns commonly used by manufacturers; especially in smooth regions, the interpolation method does not differ much among manufacturers.
Demosaicking artifacts [14]	Uses SVM classifier on feature vector composed of color features, wavelet statistics and image quality metrics.	Using feature selection algorithms or fusion techniques can enhance the classification accuracy.
Inter-pixel correlation model [17]	Models the periodic correlation between pixels as a quadratic form, uses PCA for dimensionality reduction and neural networks for classification	Sensitive to operations that change neighborhood correlation, such as, median filtering.
Proposed method	Uses probabilistic SVM classifier on features extracted by binary similarity measures, wavelet statistics and image quality metrics. Both feature and decision level fusion is used.	Good discrimination among 16 cameras. Robust to manipulations from JPEG compression, cropping and resizing to rotation, if trained appropriately.

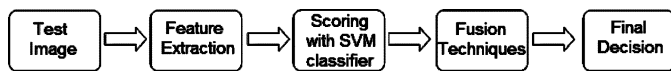


Fig. 1. Block diagram of the proposed algorithm for source camera identification.

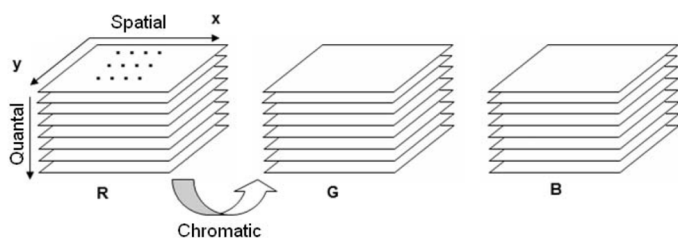


Fig. 2. Illustration of the quantal, spatial and chromatic dimensions for binary pattern similarities.

As convincing evidence, the behavior of the BSM features, the so-called the Ojala moments, is displayed in Fig. 3. The graphs in Fig. 3(a) contain the plot of the Ojala moments not only for different camera models, but also for different units of the same model. One can observe that different models have significantly different histograms while different units of the same model (e.g., Nokias in the graph) have histograms that differ in peak magnitudes. In Fig. 3(b), we observe Ojala histograms of images under various manipulations. The underlying discriminating characteristics of the Ojala histograms still persist for different models from manipulation to manipulation, albeit with some variations (notably under rotation).

Characteristics of image denoising residuals: Denoising has been a frequently used tool to extract "content-independent" parts of the image. These are components that purportedly

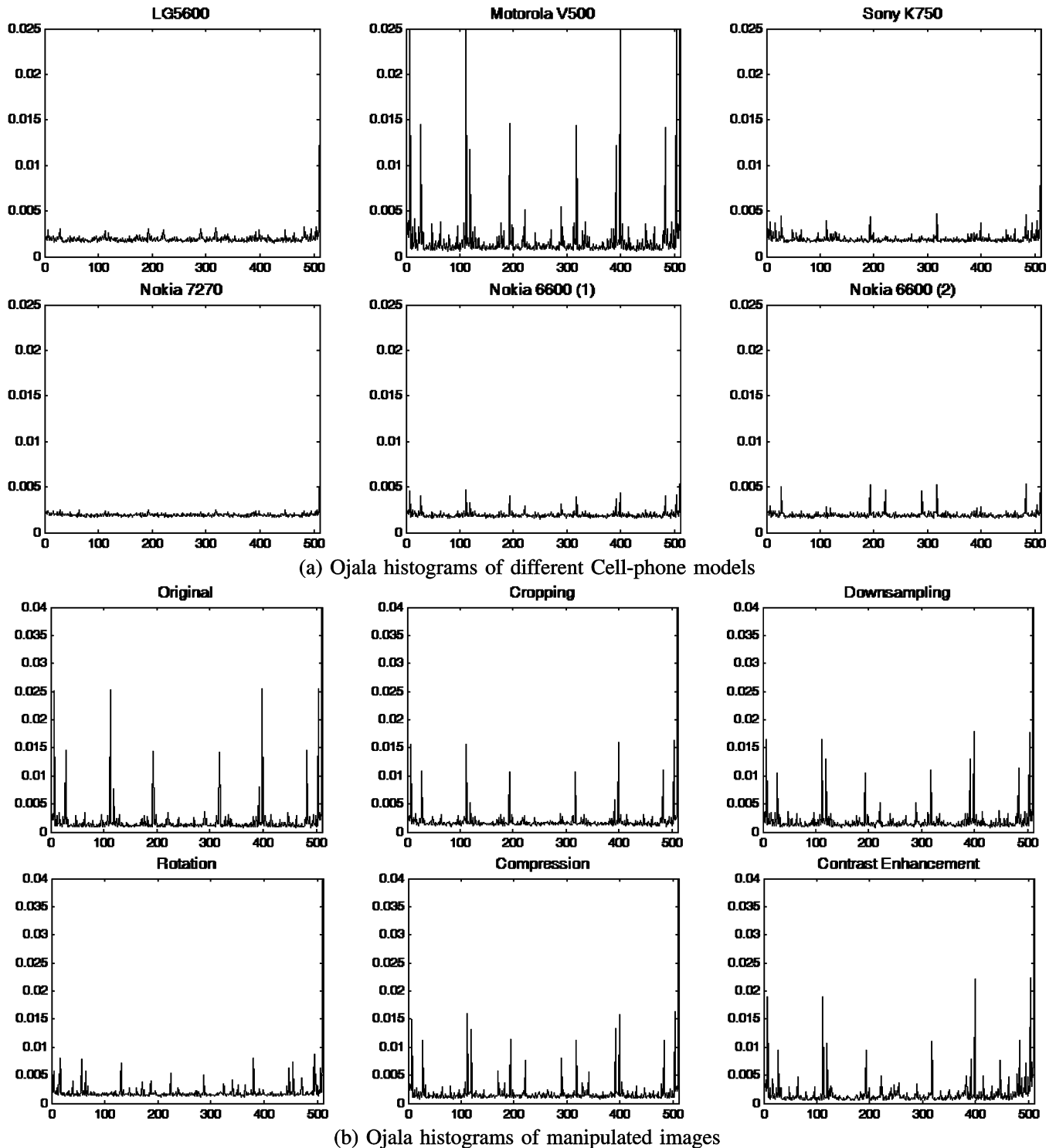


Fig. 3. Plot of 512-bin Ojala histograms for images from different cameras or for manipulated images from the same camera.

belong to imaging system noise and artifacts. The various generalized moments of denoising residual are called image-quality measures (IQM). Image-quality measures were extensively studied in [2] for steganalytic purposes. In this paper, the denoised image was obtained with a Gaussian smoothing filter ( $\sigma = 0.5$ ) and a mask of size  $3 \times 3$ , and by subtracting the denoised image from its original version. The IQM set is evaluated for each color band separately resulting in a total of 40 features (see Appendix A).

Characteristics of correlation within and between wavelet bands: These features, called high-order wavelet statistics (HOWS) in [4] and [19], have proven to be very effective in image steganalysis and tamper detection. These features are based on the decomposition of the image using separable quadrature mirror filters, which split the frequency domain into multiple scales and orientations. Features consist of the four moments (mean, variance, skewness, and kurtosis) of the wavelet coefficients over  $N$  subbands and three orientations as

well as of the norms of the optimal linear predictor residuals, resulting in 72 features overall. The HOWS features were used previously for steganalysis [19], for image tampering detection and classification [4], and for camera identification [14].

### B. Feature Selection

We ran feature selection algorithms to optimize the number of features, which is tantamount to feature fusion. In any case, the large number of initially proposed features had to be reduced for classifier design. We used the sequential forward feature selection (SFFS) algorithm, which provides reliable results at an affordable computational cost [23]. The SFFS method analyzes the features in ensembles and constructs the final set by adding to and/or removing from the current set of features until no more performance improvement is possible. The SFFS procedure can be briefly described.

- Start by choosing from the set of  $K$  features the pair of features yielding the best classification result.
- Add the most significant feature from the remaining ones.
- Remove the least significant feature from the selected set by conditionally removing features one by one, while checking the performance variations. If it improves, remove this feature and continue with another remove cycle; otherwise, do not remove this feature and try to add another one.

In the feature fusion approach, we pooled the three feature categories, namely, BSM, IQM, and HOWS, and performed the SFFS algorithm on the whole feature set in order to reduce the dimensionality of the feature vector and improve performance. The ensemble of features was reduced from 592 to 192 features; including 119 BSM, 40 HOWS, and 33 IQM features.

### C. Classification Method

We used as a classifier support vector machines (SVM) of the RBF variety. In order to set the SVM parameters, we evaluated the performance for different kernel parameters  $\gamma$  and cost parameters  $C$  by scanning over the ranges  $\gamma = [2^3, 2^2, 2^1, \dots, 2^{-15}]$  and  $C = [2^{15}, 2^{14}, 2^{13}, \dots, 2^{-5}]$ . In the final analysis,  $\gamma = 2^{-3}$  and  $C = 2^8$  have been selected as the best parameter pair on the basis of the search scheme. Since the SFFS algorithm is computationally costly, we have a particular training and test set for feature selection. Once we obtained a subset of features, we trained our SVM classifier with randomly selected images and tested our SVM classifier on the unseen half of the images. We repeated this process 100 times and overall performance was obtained by averaging classification accuracies of all different data sets. The performance changes very slightly with a standard deviation of approximately 1 or less which is evidence of the stability of the SFFS feature selection scheme over different training and testing images.

We used the continuous SVM scheme to identify cameras. Since SVMs are binary classifiers and predict only class labels, but do not yield probability information [6], we have used the LibSVM package in which SVM is extended to multiple classes yielding class probability estimates. Chang *et al.* [6] used one-against-one approach in which  $k(k - 1)/2$  binary classifiers are constructed for  $k -$  class classification. A voting strategy is considered where the final decision is reached on the basis

of the class receiving the highest vote count. In the case where two or more classes have identical votes, we break the tie by selecting the one with the smaller index.

More explicitly, given  $k$  classes of data, for an observed feature vector  $x$  and its class label  $y$ , the goal is to find the probability that  $x$  comes from the  $i$ th class; namely,  $p_i = p(y = i|x)$   $i = 1, 2, \dots, k$ . For multiclass classification, one estimates pair-wise class probabilities, defined as  $\mu_{ij} = p(y = i|y = i \text{ or } j, x)$ . From the  $i$ th and  $j$ th classes of the training set, the estimate  $\hat{\mu}_{ij}$  of the pair-wise class probabilities is found by fitting a parametric model to the posterior density functions  $\hat{\mu}_{ij} = 1/(1 + e^{\alpha\hat{J} + \beta})$ . Here,  $\alpha$  and  $\beta$  are estimated by minimizing the negative log-likelihood function using known training data and their decision values. Then,  $p_i$  can be obtained from all  $\hat{\mu}_{ij}$ 's by solving the following optimization problem:

$$\begin{aligned} \min_p \quad & \sum_{i=1}^k \sum_{j:j \neq i} (\hat{\mu}_{ij} p_i - \hat{\mu}_{ij} p_j)^2 \\ \text{subject to} \quad & \sum_{i=1}^k p_i = 1, \quad p_i \geq 0, \quad \forall i. \end{aligned}$$

Let  $p$  denote the corresponding solution. Then, define the classification rule as  $\delta = \arg \max_i [p_i]$ . Details of implementation can be found in [6] and [29].

For a given sample image, the algorithm produces a  $K$ -long confidence vector, where each component  $c(i) \in [0, 1]$ ,  $i = 1, \dots, K$  indicates the probability of the test image pertaining to the  $k$ th class. In Fig. 4, we show confidence values for each cell-phone camera model. Each graph depicts a model (say  $k$ ) the likelihood that test images are coming from that model  $k$ , given that images are indeed from that camera model (solid curves), and the likelihood that the images are captured with any other camera model except the  $k$ th one under investigation (dashed curves), namely,  $\sum_{i \neq k}^K c(i)$ . For illustrative purposes, we used a subset of  $K = 6$  of the 16 models available. The histograms are composed of the confidence scores of 100 images and the graphs have been smoothed.

### D. Decision Fusion Method

We had three categories of features (IQM, BSM, HOWS) and, therefore, three multiclass classifiers were trained. After the classifiers yield their confidence scores, the next step is to fuse them. The decision-level fusion schemes can be grouped into three types depending on which outputs are used.

- Confidence-level fusion: The experts produce a numerical value for each class, indicating the probability that the given image belongs to that class. The fusion follows by simple arithmetic rules, such as sum, product, min, max, and median rules.
- Rank-level fusion: The experts assign a rank to each class with the highest rank being the first choice. The Borda count scheme combines the rankings produced by different experts by summing the class ranks and then assigning the test image to the class that has the smallest total rank sum.
- Abstract-level fusion: The expert only outputs a unique class label by choosing the camera that receives the maximum confidence score. For a final decision, we select the class label that has the majority of votes.

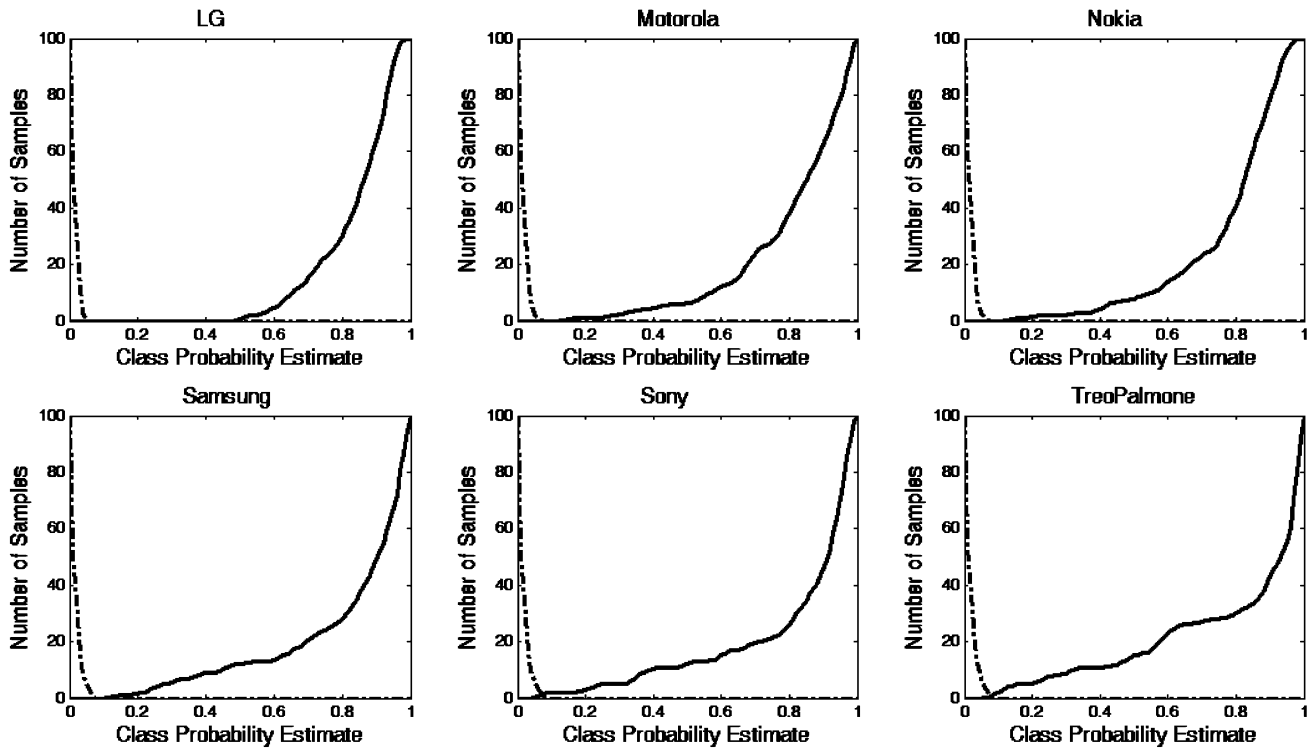


Fig. 4. Conditional histograms of the two hypotheses for cell-phone cameras. The solid lines indicate the probability that a given sample from a camera image belongs to that camera model and the dashed lines indicate the probability that it belongs to any other camera model.

In the experimental results section, we compare the results of these fusion schemes.

#### IV. EXPERIMENTAL RESULTS

We have considered  $K = 16$  models of cell-phone cameras. The six brands are Motorola, Nokia, Samsung, Sony, Treo Palmone, and LG. There are various models of these brands; in fact, we have five Nokia models, two Motorola, three Samsung, and three Sony models. Finally, notice that Nokia 6600 pair ( $N3-N4$ ) and Sony K700 pair ( $S1-S2$ ) are units of the same model. This gives us an opportunity to distinguish not only different models but different production units from the same model. In Table II, the camera models are listed along pixel resolutions and image formats.

We collected 200 images with each one of them, a total of 3200 images, with a maximum resolution of  $640 \times 480$  at daylight, auto-focus mode, and in JPEG format. Half of them (1600 images) were used for training, and classifiers were tested with the other half set of images. The images were typical shots varying from nature scenes to closeups of people.

The SFFS scheme and all feature extraction algorithms are implemented in Matlab 7.1 with an Intel E6750, 2.66-GHz processor and 2 GB of RAM. It costs approximately 50 s to extract all 592 features from one image. Although training the SVM classifiers and selecting features are time consuming, the testing (i.e., the camera identification with SVM classifier and any fusion scheme) is realized in a fraction of a second.

##### A. Classification Performance Over Small Camera Groups

Sample confusion tables from five-camera groups are given below (best, middle, worst case tables given) in Table III.

TABLE II  
MODELS OF CAMERAS TESTED AND THEIR CHARACTERISTICS

	Model	Native Resolution		Model	Native Resolution
L1	LG 5600	640x480 pixel	Sa1	Samsung D500	1.3 megapixel
M1	Motorola V3	640x480 pixel	Sa2	Samsung D600	2.0 megapixel
M2	Motorola V500	640x480 pixel	Sa3	Samsung E720	1.0 megapixel
N1	Nokia 5140	640x480 pixel	S1	Sony K700	640x480 pixel
N2	Nokia 6230	1.3 megapixel	S2	Sony K700	640x480 pixel
N3	Nokia 6600	640x480 pixel	S3	Sony K750	2.0 megapixel
N4	Nokia 6600	640x480 pixel	S4	Sony P910	640x480 pixel
N5	Nokia 7270	640x480 pixel	T1	Treo Palmone	640x480 pixel

The average performance of all 560 different three-camera groups is 98.5% with a standard deviation of 1.55. The average performance of 4368 five-camera combinations is 97.4% with a standard deviation of 1.48. The histogram of classification accuracies is shown in Fig. 5. The overwhelming majority

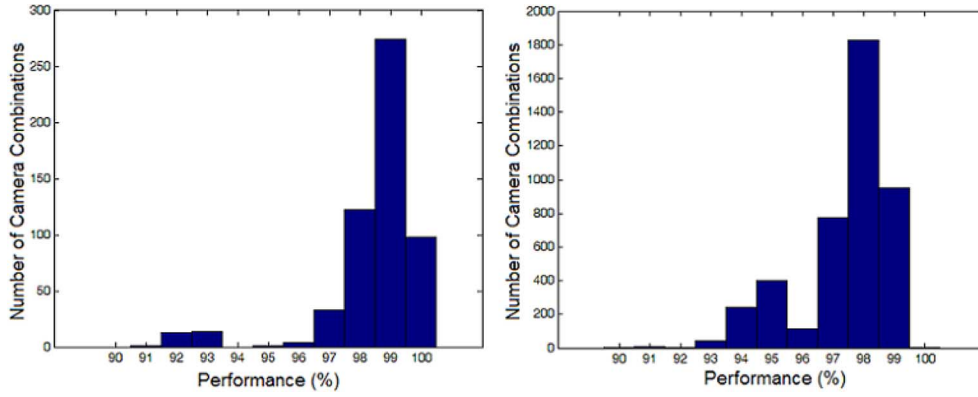


Fig. 5. Performance histograms. (a) Camera model triples using 560 different combinations. (b) Camera model quintuples from 4368 different combinations.

TABLE III  
AVERAGE PERFORMANCE SCORES OF (a) BEST GROUP,  
(b) MIDDLE GROUP, AND (c) WORST GROUP

	L1	M2	N1	S1	S3
L1	100	0	0	0	0
M2	0	100	0	0	
N1	2	0	98	0	0
S1	0	0	0	100	0
S3	0	0	0	0	100

(a) Best group 99.6%

	L1	M1	M2	N1	N2
L1	100	0	0	0	0
M1	0	88	0	0	1
M2	0	0	100	0	0
N1	1	0	0	99	0
N2	0	0	0	0	100

(b) Middle group 97.4%

	N1	N2	N3	N4	N5
N1	95	1	0	1	3
N2	1	98	0	1	0
N3	0	1	78	21	0
N4	0	1	15	84	0
N5	0	1	0	0	99

(c) Worst group 90.8%

of camera triples and quintuples is classified with very high accuracy. The outliers, that is, the low accuracy camera combinations are almost always cameras of the same brand but different models, for example, attempts to differentiate only Nokia's. One of the "worst" combinations (five Nokia's as in Table III-C) illustrates the case. Finally, we have conducted two discrimination experiments with different units of the camera model. For Sony K700 (the  $S1-S2$  pair in Table II) and Nokia 6600 ( $N3-N4$  pair), the average classification accuracy is 97.5% and 94.5%, respectively. These results have been achieved by using relatively small subsets of features. For instance, the required number of features for two-, three-, and five-camera combinations is 6, 20, and 57, respectively.

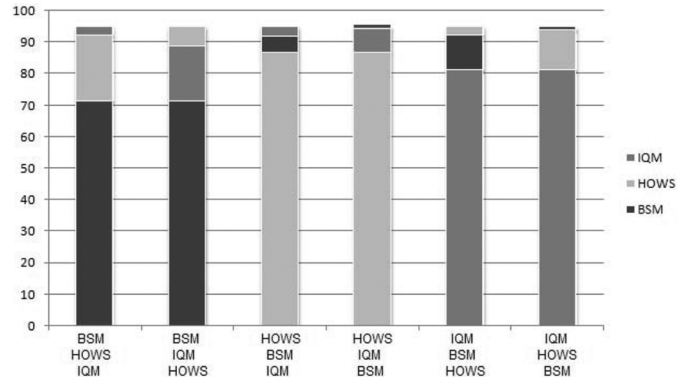


Fig. 6. Contribution of individual feature categories to overall performance for six different orderings.

### B. Classification Performance Over Large Camera Groups

In a more challenging experiment, we tried to classify camera models from a pool of 16 models. The SFFS algorithm was run for the ensemble of camera models with the SVM classifier. The feature selection algorithm resulted in 192 features out of 592 available. The contribution of each feature category to the overall performance is given in Fig. 6. Notice that the order in which the categories are introduced matters; therefore, we present all six orderings of the BSM, HOWS, and IQM categories. The first selected category contributes the lion's share of performance while the other two remain incremental. The corresponding confusion matrix is given in Table IV.

In this more crowded set, the accuracy achieved with the selected feature set was 95.1%. As could be conjectured, the different production units from the same model (e.g., Sony K700 and Sony K700) tend to be confused more often with each other compared to intermodel cases (e.g, Sony K750 and Sony P910), and the latter more with respect to the interbrand cases (e.g., Motorola versus Nokia).

### C. Feature Fusion and Feature Stability

To show the role of feature fusion, we trained and ran the algorithm with the three separate feature categories. As shown in Fig. 6, the ensemble of all 592 features achieves classification accuracy (93.5%), which is 6.1 points better compared to the best single category feature set, that is, HOWS with 87.4%.

TABLE IV  
CONFUSION MATRIX OF 16 CAMERA MODELS (PERCENTAGE OF 100 TEST IMAGES). THE OVERALL AVERAGE PERFORMANCE IS 95.1%

	L1	M1	M2	N1	N2	N3	N4	N5	Sa1	Sa2	Sa3	S1	S2	S3	S4	T1
L1	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M1	0	87	6	0	4	0	0	0	0	0	0	0	0	0	0	3
M2	0	3	97	0	0	0	0	0	0	0	0	0	0	0	0	0
N1	0	0	0	96	0	0	1	1	1	0	0	0	0	0	1	0
N2	0	0	0	0	98	1	1	0	0	0	0	0	0	0	0	0
N3	0	0	0	0	0	92	8	0	0	0	0	0	0	0	0	0
N4	1	0	0	0	0	1	98	0	0	0	0	0	0	0	0	0
N5	0	0	0	0	0	0	0	97	1	0	0	0	0	0	1	1
Sa1	0	0	0	0	0	0	0	0	99	0	1	0	0	0	0	0
Sa2	0	0	0	1	0	0	0	1	0	96	0	0	0	1	1	0
Sa3	0	0	0	3	0	0	0	0	1	0	95	0	0	0	0	1
S1	0	0	0	0	0	0	0	0	0	0	0	90	10	0	0	0
S2	0	0	0	0	0	0	0	0	0	0	0	7	93	0	0	0
S3	0	0	0	0	0	0	0	0	1	4	2	0	0	92	1	0
S4	0	0	1	1	0	0	1	0	0	0	0	0	0	0	97	0
T1	0	2	1	1	0	0	0	0	0	0	1	0	0	0	0	95

TABLE V  
PERFORMANCE OF INDIVIDUAL FEATURE CATEGORIES AND PERFORMANCE WITH ENSEMBLE AND SFFS FUSION PROCEDURES

	Ensemble Fusion		Fusion with SFFS	
	Accuracy (%)	Number of Features	Accuracy (%)	Number of Features
BSM	71.3	480	71.4	119
HOWS	87.4	72	86.8	40
IQM	81.2	40	81.2	33
Fused Feature Set	93.5	592	95.1	192

Furthermore, if we apply the SFFS scheme to the ensemble feature set, we improve the performance to 95.1%, with the additional benefit of having reduced the features down to one-third the original population, that is, from 592 to 192. We must clarify a few points on Table V. First, the feature selection starts from the pool of 592, and is not run separately for each category. Thus, the 40 HOWS features resulting from its initial population of 72 is what remains from the SFFS applied to the ensemble, and not specifically to the 72-feature HOWS category. Consequently, as in Table V, we can even observe performance drops (e.g., entire HOWS features yielding 87.4%), while after SFFS, its category-based performance drops to 86.8%. However, the important thing is the classification performance resulting from the combination of selected features in categories. Fusion with SFFS enhances the correct camera classification probability from 93.5 to 95.1. Besides its contribution to performance, we need 34 s to compute 192 features over one image while it takes approximately 50 s for the whole feature set.

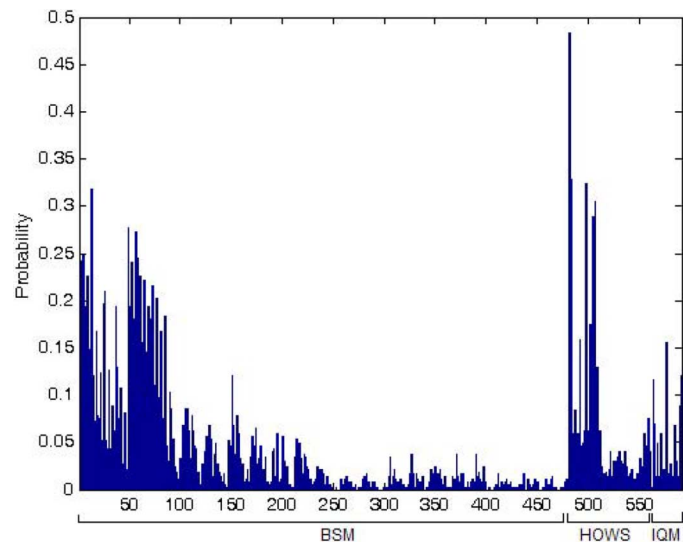


Fig. 7. Histogram of features selected by the SFFS over various configurations.

We also tested the stability of features, that is, whether we consistently obtain the same features over different combinations of camera models and over different camera populations. In Fig. 7, the probability with which each feature is selected by the SFFS scheme is shown. These figures result from averaging of histograms from several populations and combinations. First, the histograms were quite similar in appearance over populations (not shown here). When we consider the features that appear with probability of 0.1 and above, we obtain 60 features. We observe that 71.2% of these 60 features are Ojala moments (43 within 46 BSM features), ten HOWS features, and the remaining four features are IQM types.

We also tested whether the feature sets were portable from one training population to another population. For example, we

used nine cameras (L1, M1, M2, N1, N2, Sa1, Sa2, S2, and TP) to train a classifier, and then used it on threesomes of cameras. The ninesome camera group had 48 features with 98.2% performance. The same feature set, when applied to 84 three-camera groups, had an average of 98.0%, a slight decrease from the 99.72 performance when, on average, 12 features were used specific to threesomes. In Section IV-D, we investigated forensic features in more detail.

*D. Forensic Features*

In this section, we try to justify the features selected by the SFFS method out of the original 592. From the BSM category, these are distances among Ojala histograms and especially the Kullback–Leibler distance which is given in Table XI(A). The Ojala features and other BSM features are computed using the 7th and 8th bit planes. These are least dependent on the image content and, hence, are more effective features. The frequently selected HOWS features are the mean and variance of the wavelet coefficients. The highest histogram corresponds to the mean of the horizontal wavelet coefficients in the first level decomposition. Among IQM features, block spectral phase error and Czekonowsky distance between color components are the favorite features. Generally, both IQM features and BSM features are concentrated on the red and green channel of the image rather than the blue channel. Finally, we considered the number of adequate features as a function camera population. The required number of features rapidly rises to 192 for 16 cameras, while it is 73 and 74 for seven-camera combinations and nine-camera combinations, respectively. In this case, the 72 HOWS features are reduced to 40, consisting of the mean, variance, skewness, and kurtosis of all three-level wavelet decomposition and of the moments of the linear predictor residuals, but mostly at the first level of wavelet decomposition. Higher levels of wavelet decomposition have sparser prediction moments selected. The SFFS scheme selects 33 IQM features out of 40. The discarded features are generally the ones extracted from the blue channel, since red and green channels prove to be more informative. Finally among BSM features, the Ojala histograms, which are computed between different bit planes of different color channels, are very important.

One can associate these features to various processes in the camera pipeline. The BSM features can be conjectured to be sensitive to the organization of bit patterns of the seventh and eighth bit planes and possibly bear the footprints of the CFA interpolation. The HOWS features are sensitive to the moments of directed wavelet details as well as their predictability from spatial and across the band neighbors. Finally, IQM features are sensitive to the changes in the red-green channel with camera artifacts and sensor noise.

To complete the analysis, we investigated two more points. First, we wanted to see if camera identification could be conducted on gray-level images instead of using their RGB bands. The forensic procedure applied gray-level images resulting in about five percentage points lower performance compared to the setup above which extracts features separately from the three color bands. The classification accuracy with 16 camera models fell from 95.1 with color images to the average figure of 90.9 with gray-level images. Second, we wondered if the training set

TABLE VI  
CONFUSION MATRIX OF BRANDS. THE AVERAGE PERFORMANCE OF THE CASCADE IDENTIFICATION SCHEME IS 97.5% WHILE DIRECT (NONCASCADE) CLASSIFICATION PERFORMANCE IS 95.1%

	LG	Motorola	Nokia	Samsung	Sony	Treo Palmone
LG	100	0	0	0	0	0
Motorola	0	100	2	0	0	3
Nokia	0	0.2	98	1.4	0.4	2
Samsung	0.3	0.3	3	95.3	0.3	0.6
Sony	0	0	1	1.75	97	0.25
Treo Palmone	0	2	2	0	0	96

was adequate for the classifiers. Recall that in all of our work, we employed 100 images for training and 100 for testing. Although at around 100, the performance seems to reach a saturation point, it increases slightly if we almost double the set of training images. In other words, one obtains 95.1% correct classification with 100 training images and 96.8 with 190 training images. This performance differential, however, can be recovered with decision fusion, as detailed in Section IV-E.

*E. Roles of Brand-Model Hierarchy and of Decision Fusion*

We experimented with a cascade identification scheme where the first tier identifies the camera brand and the second tier, the model within that brand. There were six camera brands according to manufacturers as in Table II. The average classification accuracy of brands using a full set of features is 97.5%; the individual brand performances are given in Table VI. Once the brand is identified, one can proceed with the identification of the camera model. In this scheme, the individual identification performance of each model generally increases; especially for Nokia 7270, Samsung D500, Sony K750, and Sony P910, performance reaches to 100%, and the score of Motorola V3 jumps by 10%. On the other hand, for the Nokia 5140 and SonyK700, identification performance drops to 77.8% and 86.2%, respectively. The average performance of the cascade identification scheme is 97.5% while direct classification performance is 95.1%.

An alternate fusion method is the decision fusion technique, where we combine the decisions from various classifiers. These classifiers were again based on feature categories, namely, BSM, IQM, and HOWS. In Table VII, we present the classification accuracy for seven different decision fusion schemes. The highest accuracy is achieved with score-level fusion. Here, for a given test image, we obtain its normalized confidence scores for each feature category, and then fuse these scores by using the product rule.

*F. Camera Classification Under Image Manipulations*

Images presented to a camera identifier may have been subjected to a host of manipulations, malicious or innocent, such as cropping, contrast enhancement, or JPEG compression.

TABLE VII  
COMPARISON OF DIFFERENT DECISION-LEVEL FUSION SCHEMES

Combination Schemes		Performance (%)
Score-level Fusion	Sum Rule	96.7
	Mean Rule	96.6
	Product Rule	97.5
Abstract-level Fusion	Majority Voting	94.4
Rank-level Fusion	Borda Count	95.2
Feature-level Fusion	SFFS	95.1
Without SFFS pruning		93.5

TABLE VIII  
IMAGE MANIPULATION TYPES AND THEIR PARAMETERS

	Parameters								
Cropping(%)	10	20	30	40	50				
Downsamp.(%)	1	5	10	15	25	30	40	50	
Rotation (deg)	1	3	5	10	15	25	30	45	
JPEG Comp.	90	80	75	60	50	40			
Contrast Enh.	1	5	10	15	25				
Sharpen									Photoshop Default

The types of manipulations the images were subjected to and their strength parameters are listed in Table VIII, where 33 different parameter settings are listed. Notice that there are several instances of “heavy manipulation,” (e.g., 50% cropping, 50% downsampling, rotations above 5°, etc.). Since each image is subjected to all of the 33 type or strength of manipulations, the total number of training/test images is boosted to  $9 \times 200 \times (1 + 33) = 61,200$ . Notice that this part of the experiment was conducted on a subset of camera models, namely, on the nine LG, Motorola, Nokia, and Sony Ericsson brands and their models.

We have conducted two classification experiments, called, respectively, the naïve and the informed set.

- Naïve classifier: The classifier is trained only with the original “unmanipulated” set of 100 images per camera, which is effectively the classifier designed in Section IV-C. However, it has to identify cameras based on variously manipulated images according to Table VIII. In this case, there are overall  $100 \times 34 \times 3400$  test images per camera. We do not use manipulated versions of the 100 nonmanipulated training images while testing.
- Informed classifier: The classifier is denoted so because it is trained with the original and the manipulated sets of images. In this case, the database is partitioned into 3400 training and 3400 test images per camera. In manipulation experiments, we handled the identification problem by means of decision fusion techniques; especially the product rule.

TABLE IX  
IDENTIFICATION PERFORMANCE IN GROUPS OF THREE AND NINE FOR NAÏVE AND INFORMED CASES

	Originals only	Naïve Train: Original Test: Original& manipulated	Informed Train/Test: Original& manipulated
Three-somes	99.1	85.3	95.7
Nine-some	96.8	72.5	89.1

We observed that similar to the case of watermark detection, camera classification performance suffers heavily under manipulations, especially geometrical attacks, such as rotation and downsampling, while they fare much better under gray-level manipulations, such as contrast enhancement. As illustrated in Fig. 3(b), Ojala histograms are heavily affected by geometrical attacks. Our results also showed that our method is robust to JPEG compression until factor 60 and to the most common geometrical operation, that is, cropping. We refrain from giving detailed performance tables under for each attack strength and manipulation type, but we simply quote the average performance figure, which for the naïve classifier falls down to 72.5%.

To recuperate this loss of performance, we retrained a classifier with the set of  $9 \times 3400 = 30,600$  images, which corresponds to the design of the so-called informed classifier. In other words, we gave the classifier a chance to see all sorts of manipulations along with the original images. The performance was, as a consequence, significantly improved, catching up almost with the identification performance run on original images only. These average performances are summarized in Table IX. In this case, the performance of the informed classifier for nine-tuple under all manipulations increases by more than 15 points from 72.5 to 89.1%.

In Fig. 8, selected performance results are presented for three-some (averaged over 84 combinations) and ninesome cellular cameras. The goal here is to illustrate on the one hand the performance drop with the increase in the strength of the manipulation, and, on the other hand, how the original performance is recovered when classifiers are retrained using original plus manipulated images. Notice that the recovery after signal-processing attacks [Fig. 8(a) and (b)] is much more feasible compared to recovery after geometrical attacks [Fig. 8(c) and (d)]. We show the improvement of the informed classifier *vis-à-vis* naïve classifier in a bar graph in Fig. 9, where the lower (darker) part of the bar indicates the naïve case performance and the lighter upper part of the bar indicated the informed case.

### G. Digital Cameras Versus Cell-Phone Cameras

In this section, we extend the forensic analysis to digital cameras, that is, consumer cameras. Obviously, there are differences in degrees, if not in type between digital cameras and cell-phone cameras. In other words, while their respective image-processing pipelines are similar, there are significant differences in quality. It would be easier to point out the limitations of cell-phone cameras *vis-à-vis* their digital counterparts.

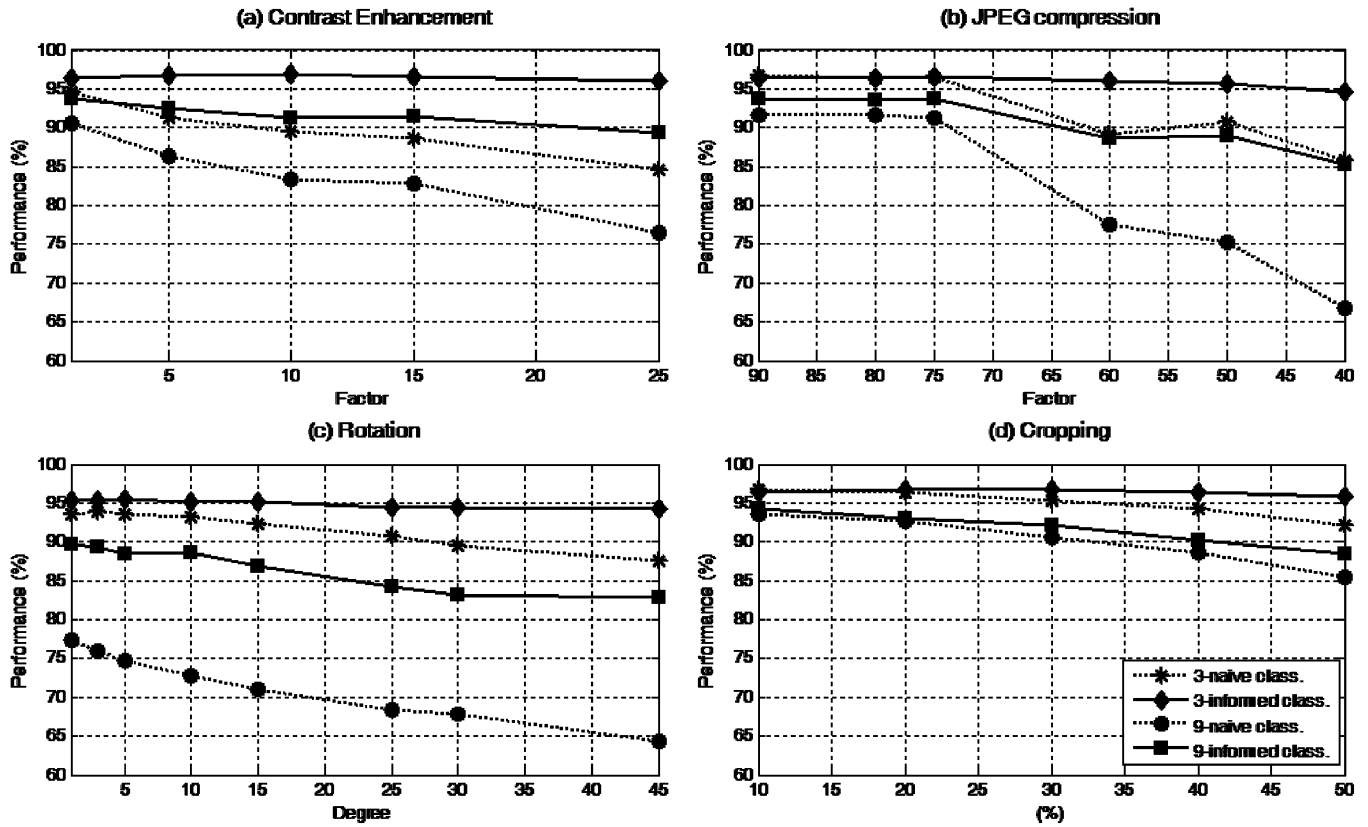


Fig. 8. Correct identification performance curves *vis-à-vis* the strength of the manipulation: comparison of the naïve and the informed cases.

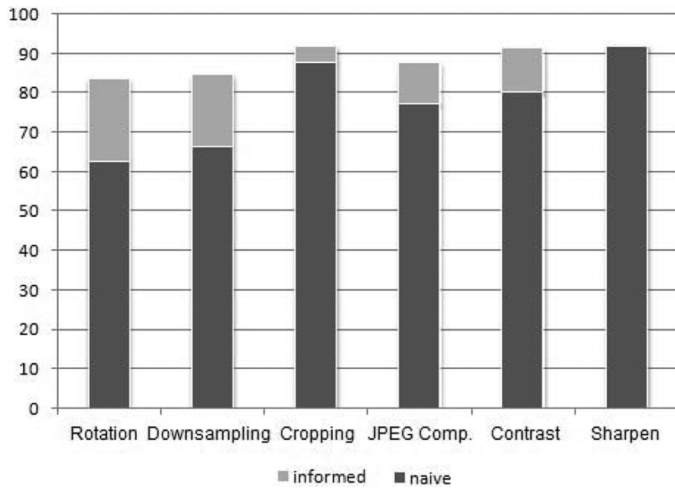


Fig. 9. Performance under image manipulations for naïve and informed cases (nine cameras).

The cell-phone cameras result in lower quality images due to several factors. First, they have lower resolution. Second, they have a fixed *f*/number (focal length of the lens) which limits the lighting conditions and ultimately the quality of images. Cell-phone cameras, by their compact nature, are constrained to have very small aperture stops and focal lengths. In contrast, cameras allow the aperture size, hence *f*/number, to be adjusted, thus providing more flexibility to the photographer. Third, the cell-phone camera flashes are not as robust as in digital cameras or absent altogether due to the power constraints. Light sensitivity of the sensor type directly affects shutter speeds which are

related to the blur in action shots. Fourth, digital and cell-phone cameras also differ in analog–digital conversion (ADC). In the former ones, a 12-b ADC is typically used while that of the cell phone is limited to 10 b. In conclusion, electronic hardware makes the difference between these two camera categories.

In order to test the performance of our algorithm on both variety cameras, we selected four digital camera types, namely, Canon S100, Canon S110, Canon S200, and one Nikon E2100. The digital camera database is composed of 150 images taken by each camera at a resolution of  $1600 \times 1200$  pixels. The identification performance with digital cameras is 95% and this accuracy is upgraded to 96.5% by decision fusion techniques, which is quite close to the average performance foursomes of cell-phone cameras, 98.7% (i.e., three of the same brand and one different brand of cell phone). An interesting issue was whether the digital and cell-phone cameras generated the same forensic features. We found that only 49.5% of their features were common; hence, they were not interchangeable. For example, the digital camera classification with cell-phone features dropped from 95% to 75.5%. The noninteroperability of feature sets is obviously due to widely different quality of images between the two camera categories. CFA-related footprints are more prominent in digital cameras while cell-phone cameras have higher noise contamination due to the quality differential between complementary metal–oxides semiconductor (CMOS) and CCD technologies. Therefore, BSM features, especially Ojala histograms, are more effective in digital cameras due to CFA-related patterns. On the other hand, IQM and HOWS features are more distinctive in cell-phone identification as they are directly computed over image residuals (noise).

### H. Comparison With Existing Methods

It is difficult to make a full comparison with the methods in the literature since experimental protocols, feature sets, training/testing image sizes, and number of cameras all differ. In most of the previous works, the purpose was to classify digital cameras, which are high-qualified cameras compared to cell-phone cameras. However, in comparing our method with its closest correlations, we make the following observations.

The methods closely related to ours are the demosaicking artifact-based ones, such as in [5], [14], [17], and [28]. Our method differs from theirs in that we have the additional category of binary similarity measure features and that we employ the decision-level fusion. For example, in their initial work, Kharazzi *et al.* achieved 88.02% accuracy among five digital cameras [14]. In a more recent work in the same vein Bayram *et al.* [5] achieved 96% for three camera case, while we average 98.5. In [28], a similar approach as in [14], is applied to the camera data set consisting of a mixture of cell-phone and digital cameras; the identification performance is 98.35% among three cameras and three cell phones. In the pure camera set, their identification accuracy is 93.05% among four digital cameras while we achieve 96.5% and 98.7% performance in identification of cameras and cell phones, respectively. They also identify the camera model among seven different cameras of the same brand with 95.1% accuracy, while we average 98.5 for different brands. In [17], Long and Huang have estimated the CFA interpolation method and used feedforward back propagation neural networks instead of SVM for classification. Their average accuracy for four cameras is 98.25%, which is on a par with ours, but their experimentation is limited to only JPEG manipulation. Swaminathan *et al.* [27] in a nine-brand experiment achieved a performance of 90% and 86% for the 19-class problem whereas we obtain 97.5% for both six camera brands and 16 camera models. In all fairness, we must note that the camera set in [27] consists of 19 digital cameras; hence, it may represent a more challenging repertoire. In [8], lens radial distortion is used for identification via the same features as in [14] and achieved 89.2% accuracy for five cameras. In Table X, we compared the performance of the proposed method to the existing methods by considering the number of training/testing samples and cameras used in the experiments.

### V. CONCLUSIONS AND FUTURE WORK

We have shown that with a judicious combination of forensic features, it is possible to identify the source cellular-phone camera model of an image with satisfactory accuracy. The following conclusions can be drawn.

- Useful forensic features result from a combination of lower-order bit plane patterns (BSM), of moments of wavelet components (HOWS) and of certain functions image denoising residuals (IQM).
- The combination of these features via ensemble fusion or fusion consequent to a feature selection scheme is beneficial.
- Decision fusion with a product rule, where classifiers are based on feature categories, enhances the performance. A total of 7.8 point improvement is obtained *vis-à-vis* a best single category classifier (HOWS).

TABLE X  
PERFORMANCE COMPARISON WITH EXISTING METHODS

Method	Number of training/test samples	Number of cameras	Accuracy (%)
CFA interpolation method [27]	125/75	19	86
Demosaicking artifacts [14]	60/90	5	88.02
Lens Radial Distortion [8]	40/100	5	89.2
Demosaicking artifacts [28]	60/90	4	93.05
CFA interpolation method [5]	45/95	3	96
Proposed method	100/100	7	Raw=97.3 SFFS=97.8 Fusion=98.5
Proposed method	100/100	9	Raw=95.7 SFFS=96.6 Fusion=98
Proposed method	100/100	16	Raw=93.5 SFFS=95.1 Fusion = 97.5

- The feature sets are stable over camera combinations with a given population size since the performance fluctuations have 1.5 standard deviation; they are also fairly stable over different camera population sizes (e.g., between threesomes and ninesomes).
- The proposed algorithm works satisfactorily with either digital cameras and cell-phone cameras, though their feature sets do not seem to be interchangeable.
- The camera-identification scheme is robust against image manipulations, provided the classifier has been trained with sample images subjected to the attacks.
- There are no methodological differences between cell-phone and digital cameras. The only difference is the discriminative quality of the features and their types.

We plan to extend this work along two avenues using a much larger scale of units and varieties of cameras: 1) we will study forensic classifiers with mixtures of cell-phone and digital cameras and 2) we will investigate identification potential of units of the same model and brand. Our experiments with the limited number of units (two Sony K700 and two Nokia 6600) and the more detailed study indicate that there is a considerable potential for [7] model-independent unit identification.

TABLE XI

(a) OJALA FEATURES and (b) THE WEIGHTING PATTERN OF THE NEIGHBORS IN THE COMPUTATION OF OJALA SCORE. FOR EXAMPLE, THE SCORE BECOMES  $S = 2 + 4 + 8 = 14$  IN THE EXAMPLE WHERE E, N, NE BITS ARE 1 AND ALL OTHER BITS ARE 0

Features	Definiton
Ojala Kullback-Leibler Distance	$-\sum_{n=1}^N S_n^7 \log \frac{S_n^7}{S_n^8}$
Ojala Mutual Entropy	$-\sum_{n=1}^N S_n^7 \log S_n^8$
Ojala Absolute Histogram Difference	$-\sum_{n=1}^N  S_n^7 - S_n^8 $
Ojala Minimum Histogram Difference	$-\sum_{n=1}^N \min(S_n^7, S_n^8)$

(a)

1	2	4
128	256	8
64	32	16

(b)

APPENDIX A

BINARY SIMILARITY MEASURES

Consider the  $3 \times 3$  neighborhood of a binary pixel in a bit plane and obtain 512-bin histograms according to the configuration of these nine bit patterns. These histograms, also called Ojala [20] histograms, are computed as  $S = \sum_{i=0}^7 x_i 2^i$  by weighting the eight directional neighbors as shown in Table XI(b). Here, we define  $s_n^7$  as the count of the  $n$ th histogram bin in the seventh bit plane and  $s_n^8$  as the corresponding one in the eighth plane. After normalizing these 512-bin histograms, we compute four Ojala features over the spatioquantal and spatiochromatic patterns between bit planes; resulting in a total of  $4 \times 9 \times 2 = 72$  features.

APPENDIX B

IMAGE-QUALITY MEASURES

The second model of features, image-quality measures, was extensively studied in [2]. We present here one of the most distinctive measure for illustrative purposes. The Czekonowsky distance gives a metric useful to compare vectors with strictly nonnegative components, as in the case of color images

$$M = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N \left( 1 - \frac{2 \sum_{k=1}^3 \min(C_k(i, j), \hat{C}_k(i, j))}{\sum_{k=1}^3 (C_k(i, j) + \hat{C}_k(i, j))} \right).$$

In this paper, we used a Gaussian smoothing filter ( $\sigma = 0.5$ , mask size =  $3 \times 3$ )  $H(m, n) = Kg(m, n)$  where  $g(m, n) = (2\pi\sigma^2)^{-1} \exp -(m^2 + n^2)/2\sigma^2$  is the 2-D Gaussian Kernel and  $K = (\sum_m \sum_n |g(m, n)|)^{-1/2}$  is the normalizing constant. In these definitions,  $C_k(i, j)$  corresponds to the  $(i, j)$ th

TABLE XII  
IMAGE-QUALITY METRIC SET

Minkowsky Metric $\gamma = 2$	Czekonowsky distance
Minkowsky Metric $\gamma = 1$	Spectral Phase
Normalized Cross Correlation	Spectral Magnitude
Structural Content	Weighted Spectral Distance
Normalized Absolute Error (HVS)	Median Block Spectral Magnitude
HVS Based L2	Median Block Spectral Phase
Laplacian Mean Square Error	Median Block Weighted Spectral Distance

pixel of the  $k$ th band of a color image and  $\hat{C}_k$  denotes the denoised version of the corresponding  $k$ th band of  $M \times N$  color image. Denoising is employed on the image to obtain a reference image to calculate the metric. The IQM set is composed of 14 features, given in Table XII. These features are computed separately for each color channel, except for the Czekonowsky distance.

REFERENCES

- [1] J. Adams, K. Parulski, and K. Spaulding, "Color processing in digital cameras," *IEEE Microw.*, vol. 18, no. 6, pp. 20–29, Nov./Dec. 1998.
- [2] I. Avcibas, N. Memon, and B. Sankur, "Steganalysis using image quality metrics," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 221–229, Feb. 2003.
- [3] I. Avcibas, M. Kharrazi, N. Memon, and B. Sankur, "Image steganalysis with binary similarity measures," *J. Appl. Signal Process.*, vol. 17, pp. 2749–2757, 2005.
- [4] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection," *J. Electron. Imaging*, vol. 15, no. 4, Dec. 2006.
- [5] S. Bayram, H. T. Sencar, and N. Memon, "Improvements on source camera-model identification based on CFA interpolation," in *Proc. Working Group 11.9 Int. Conf. Digital Forensics*, FL, 2006.
- [6] C. C. Chang and C. J. Lin, LIBSVM: A Library for support vector machines 2001. [Online]. Available: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [7] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Security Forensics*, vol. 3, no. 1, pp. 74–90, Mar. 2008.
- [8] K. S. Choi, E. Y. Lam, and K. Y. Wong, "Automatic source identification using the intrinsic lens radial distortion," *Opt. Express*, vol. 14, no. 24, pp. 11551–11565, Nov. 2006.
- [9] E. Dirik, H. T. Sencar, and N. Memon, "Source camera identification based on sensor dust characteristics," in *Proc. Signal Processing Applications Public Security Forensics*, Apr. 11–13, 2007, pp. 1–6.
- [10] Z. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, "Methods for identification of images acquired with digital cameras," in *Proc. Enabling Technologies for Law Enforcement and Security*, Feb. 2001, vol. 4232, pp. 505–512.
- [11] H. Gou, A. Swaminathan, and M. Wu, "Robust scanner identification based on noise features," *Proc. Security, Steganography, Watermarking of Multimedia Contents*, vol. 6505, pp. 65050S–65050S, 2007.
- [12] N. Khanna, A. K. Mikkilineni, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Scanner identification using sensor pattern noise," presented at the SPIE Int. Conf. Security, Steganography, Watermarking of Multimedia Contents IX, San Jose, CA, Jan. 2007.
- [13] N. Khanna, A. K. Mikkilineni, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Forensic classification of imaging sensor types," presented at the SPIE Int. Conf. Security, Steganography, Watermarking Multimedia Contents IX, San Jose, CA, Jan. 2007.
- [14] M. Kharrazi, H. T. Sencar, and N. Memon, "Blind source camera identification," in *Proc. Int. Conf. Image Processing*, 2004, vol. 1, pp. 709–712.

- [15] J. Kittler and F. M. Alkoot, "Sum versus vote fusion in multiple classifier systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 1, pp. 110–115, Jan. 2003.
- [16] K. Kurosawa, K. Kuroki, and N. Saitoh, "CCD fingerprint method—Identification of a video camera from videotaped images," in *Proc. Int. Conf. Image Processing*, Kobe, Japan, Oct. 1999, pp. 537–540.
- [17] Y. Long and Y. Huang, "Image based source camera identification using demosaicking," in *Proc. IEEE 8th Workshop Multimedia Signal Processing*, Oct. 2006, pp. 419–424.
- [18] J. Lucas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.
- [19] S. Lyu and H. Farid, "Steganalysis using higher-order image statistics," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 1, pp. 111–119, Jan. 2006.
- [20] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognit.*, vol. 29, pp. 51–59, 1996.
- [21] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.
- [22] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of re-sampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.
- [23] P. Pudil, F. J. Ferri, J. Novovicov, and J. Kittler, "Floating search methods for feature selection with non-monotonic criterion functions," in *Proc. 12th IEEE Int. Conf. Pattern Recognition*, 1994, vol. 2, pp. 279–283.
- [24] P. H. A. Sneath and R. R. Sokal, *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. San Francisco, CA: Freeman, 1973.
- [25] "Special issue on data hiding," *IEEE Trans. Signal Process.*, vol. 51, no. 4, p. 897, Apr. 2003.
- [26] Y. Sutcu, S. Bayram, H. T. Sencar, and N. Memon, "Improvements on sensor noise based source camera identification," *Proc. IEEE Int. Conf. Multimedia Expo.*, pp. 24–27, 2007.
- [27] A. Swaminathan, M. Wu, and K. J. R. Liu, "Non-intrusive component forensics of visual sensors using output images," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 1, pp. 91–106, Mar. 2007.
- [28] M. J. Tsai, C. L. Lai, and J. Liu, "Camera/mobile phone source identification for digital forensics," in *Proc. IEEE Int. Conf. Acoustic, Speech Signal Processing*, 2007.
- [29] T. F. Wu, C. J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *J. Mach. Learning Res.*, pp. 975–1005, 2004.



**Oya Çeliktutan** received the B.S. degree in electronics engineering from Uludag University, Bursa, Turkey, in 2005 and is currently pursuing the M.S. degree in signal processing and communications at Boğaziçi University, İstanbul, Turkey.

Currently, she is a Teaching Assistant at Boğaziçi University. Her research interests include multimedia forensics, pattern recognition, computer vision, and image processing. She is now working on automatic 2-D facial feature extraction.



**Bülent Sankur** (SM'90) received the B.S. degree in electrical engineering from Robert College, İstanbul, Turkey, and received the M.Sc. and Ph.D. degrees from Rensselaer Polytechnic Institute, Troy, NY.

Currently, he is with the Department of Electrical-Electronic Engineering at Boğaziçi (Bosphorus) University, İstanbul. His research interests are in the areas of digital signal processing, image and video compression, biometry, cognition, and multimedia systems. He is the Founder and Leader of the Image and Signal Processing Laboratory at Boğaziçi University, and has been serving in various industrial consulting tasks. He has held visiting positions at the University of Ottawa, Ottawa, ON, Canada; Technical University of Delft, Delft, The Netherlands; and Ecole Nationale Supérieure des Télécommunications, Paris, France.

Dr. Sankur was the Chairman of International Conference on Telecommunications (ICT'96) and of The European Conference on Signal Processing (EU-SIPCO'05) as well as Technical Chairman of the International Conference of Acoustic, Speech, and Signal Processing (ICASSP'00).



**İsmail Avcıbaşı** (M'02) received the B.Sc. and M.Sc. degrees in electronics engineering from Uludag University, Bursa, Turkey, in 1992 and 1994, respectively, and the Ph.D. degree in electrical and electronics engineering from Boğaziçi University, İstanbul, Turkey, in 2001.

Currently, he is an Associate Professor with the Electronics Engineering Department, Başkent University, Ankara, Turkey. His research interests include image processing, data compression, information hiding, and multimedia communications.