Received March 8, 2013, accepted April 6, 2013, published May 10, 2013.

Digital Object Identifier 10.1109/ACCESS.2013.2260814

Information Forensics: An Overview of the First Decade

MATTHEW C. STAMM (MEMBER, IEEE), MIN WU (FELLOW, IEEE), AND K. J. RAY LIU (FELLOW, IEEE)

Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, USA

Corresponding author: M. C. Stamm (mcstamm@umd.edu)

ABSTRACT In recent decades, we have witnessed the evolution of information technologies from the development of VLSI technologies, to communication and networking infrastructure, to the standardization of multimedia compression and coding schemes, to effective multimedia content search and retrieval. As a result, multimedia devices and digital content have become ubiquitous. This path of technological evolution has naturally led to a critical issue that must be addressed next, namely, to ensure that content, devices, and intellectual property are being used by authorized users for legitimate purposes, and to be able to forensically prove with high confidence when otherwise. When security is compromised, intellectual rights are violated, or authenticity is forged, forensic methodologies and tools are employed to reconstruct what has happened to digital content in order to answer who has done what, when, where, and how. The goal of this paper is to provide an overview on what has been done over the last decade in the new and emerging field of information forensics regarding theories, methodologies, state-of-the-art techniques, major applications, and to provide an outlook of the future.

INDEX TERMS Information forensics, tampering detection, multimedia fingerprint, anti-forensics.

I. INTRODUCTION

Every technology has its time. In the 1970's, advances in VLSI led to a new high-tech industrial revolution allowing electronic systems to be built smaller and run faster. With that as a foundation, we have witnessed since the 1980's the worldwide development of high-speed networking and wireless infrastructure. Once communication and computing infrastructure offered enough bandwidth and computational power for broadband applications, the development of multimedia compression algorithms and systems took off in the 1990's. This led to many well known standards such as JPEG, MPEG-1/2/4 and H.26x. After this came the need to search and organize content as it proliferates all over the Internet. Content search and retrieval efforts such as MPEG-7 became the focusing point of the R&D community, and commercial giants of web search such as Google and Yahoo emerged. This path of technological evolution has naturally led to a critical issue that must be addressed next, namely, to have security and forensic technologies to ensure content, devices, and intellectual properties being used by authorized users for authorized purposes and to be able to gather solid forensic evidence to hold culprits accountable when otherwise.

The widespread adoption of digital content over traditional physical media such as film has given rise to a number of new information security challenges. Digital content can be altered, falsified, and redistributed with relative ease by adversaries. This has important consequences for governmental, commercial, and social institutions that rely on digital information. To secure communication infrastructure and prevent unauthorized access to digital information, many cryptographic encryption and authentication techniques have been developed. While a large amount of effort has been devoted to developing these information security measures, their protection usually ends once the content is delivered and decrypted. This is problematic, since there is little control on how digital information is used or processed after delivery. Furthermore, these techniques cannot prevent multimedia content from being manipulated or falsified before it is cryptographically encrypted or signed.

In many important real world scenarios, digital information originates from an unknown or untrusted source. When this happens, a forger or information attacker can easily manipulate digital content such as images or video to create perceptually realistic forgeries. Encryption cannot ensure that the information being distributed is unaltered and authentic

in these situations. Before multimedia information of this nature can be trusted, several questions must often first be answered such as: what is the true origin of this content? How has it been processed? Is this content authentic or has it been manipulated? Compared with the efforts made to ensure the secure and trusted delivery of information, research approaches that are aimed at answering these questions are still relatively new.

In response to the increasing need to verify the trustworthiness of multimedia content, the field of *information forensics* was born. Information forensics is concerned with determining the authenticity, processing history, and origin of digital multimedia content with no or minimal reliance on side channels other than the digital content itself. It further aims at reconstructing who, when, how, and what has been done to the content. When information goes through various devices and processing, there are inherent traces left from each processing step. These traces are referred to as *intrinsic fingerprints* and are essential for forensic analysis. There have been many forensic techniques to identify manipulation by exploiting imperceptible traces that intrinsically arise in multimedia content during acquisition and processing.

Meanwhile, there are also extrinsically generated security measures such as digital watermarks. These security measures are often embedded into information content through some kind of data embedding technique that is imperceptible to users. We refer to these traces as *extrinsic fingerprints*. Earlier work on extrinsic techniques in the late 1990s mainly concerned with embedding information to demonstrate copyright or verify host media data's integrity. The recent decade has seen active research on extending traditional robust watermarking to embed traceable fingerprints that can identify individual copies of media content or link the copy to specific unit of acquisition device, and sustain multiple adversaries' collaboration attacks.

These notions of ''intrinsic'' vs. ''extrinsic'' fingerprints or signatures were first coined in the literature by an interdisciplinary team at the Purdue University for electrophotographic printers [1]. Leveraging their expertise in the imaging and mechanical process of printers, they took a pioneering view of the banding artifacts of printers and treated them as an ''intrinsic'' signature of the printer that can be identified by appropriate image analysis techniques. Further, the group strategically amplified and modulated the banding to embed additional information as an ''extrinsic'' signature to encode such useful information as the date and time that a document was printed. The forensic value and use of extrinsic fingerprints was also recognized in traitor-tracing work, and the term ''multimedia forensics'' first appeared in the title of a group-oriented embedded fingerprinting paper [2]. A search on Google Scholar using the term ''information forensics'' returns about 26,000 entries published from 2001 to early 2013.

Although the field of information forensics is still young, many forensic techniques have been developed to detect forgeries, identify the origin, and trace the processing history of digital multimedia content. This paper provides an overview of information forensics research since the field began roughly a decade ago. We begin by discussing forensic techniques designed to identify forgeries and trace the processing history of digital image, video, and audio files. Next, we examine the device-specific fingerprints left by digital image and video cameras along with forensic techniques that use these fingerprints to identify the source of digital multimedia files. We also discuss environmental fingerprints and forensic traces such as those introduced by the electrical power network.

The study of information forensics cannot be complete without a balanced view from the perspective of an adversary. Because of this, we proceed by discussing the recent development of anti-forensic countermeasures designed to fool forensic techniques and examining the resulting adversarial dynamics between a forensic investigator and an information forger. After this, we review embedded fingerprints for traitor tracing, then discuss social factors and behavior dynamics in forensics. In fact, an adversary has all the incentives to erase or degrade the traces of evidence to an undetectable level or forge new traces to lead to a wrong or ambiguous conclusion. We conclude with a few final thoughts and a discussion of future directions.

II. DETECTION OF TAMPERING AND PROCESSING OPERATIONS

In many scenarios, multimedia content from an unknown or untrusted source contains important information. For example, an image or video released by a hostile foreign government may depict an event with significant political or military consequences. Before the events depicted in this media can be trusted, it is first necessary to identify any processing the multimedia file has undergone and determine its authenticity. Since it is unlikely that the multimedia content would have had security measures embedded in it before any processing occurred, researchers have developed a variety of forensic techniques that operate without relying on extrinsic security measures. These techniques operate by exploiting properties of the digital content itself.

Based on the manner in which they operate, forensic techniques designed to trace a multimedia file's processing history, detect manipulation, and identify forgeries can be grouped into five basic categories. These include statistical classifiers that use machine learning algorithms trained on a broad set of image statistics, techniques that search for inconsistencies in device fingerprints, techniques that detect manipulation-specific fingerprints, methods that make use of compression and coding fingerprints, and techniques that search for physical inconsistencies in the multimedia content. While no single forensic approach can identify all forms of manipulation, if used together, these techniques can reliably detect many types of forgeries and a wide variety of processing operations. Here, we provide an overview forensic techniques in each of these five categories.

A. STATISTICAL CLASSIFIERS

The manipulation or falsification of multimedia information inevitably leads to changes in its statistical properties. In some cases, analytical examination of the manipulation or forgery process can allow researchers to identify the specific statistical changes that arise when certain processing operations are performed. If this is possible, these unique changes can be used as intrinsic fingerprints when performing forensic analysis. In many cases, however, it is extremely difficult to identify these intrinsic fingerprints. One approach to overcoming these difficulties is to search for changes in a representative set of statistical features extracted from a multimedia file.

Several forensic techniques, which can be broadly grouped together as statistical classifiers, adopt this approach to identifying inauthentic multimedia content. Instead of developing a set of features targeted at detecting specific manipulation fingerprints, these techniques make use of general statistical features that are likely to change when a multimedia file is altered or falsified. These forensic techniques use powerful tools from machine learning such as support vector machines (SVMs) or neural networks to identify the specific changes in these features that correspond to manipulation. Before classification can be performed, these techniques must be trained using features extracted from both unaltered and falsified or manipulated multimedia content.

The performance of these algorithms hinges largely on the selection of an appropriate set of features for discriminating between each class. Features developed to perform steganalysis can successfully be used to identify image manipulation [3]. By constructing a joint feature set of steganalysis features consisting of image quality measures, higher order wavelet statistics, and binary similarity measures, machine learning algorithms can be used to discriminate between manipulated and unaltered images [4]. These features can also be used to train targeted detectors to identify images manipulated by rescaling, rotation, contrast or brightness changes, blurring, or sharpening. It is worth noting that steganalysis can be viewed as a specific type of forensic analysis, with the goal to determine whether or not there is secondary data embedded in the primary signal (also known as host signal or cover signal). Although we will not elaborate on steganalysis in the interest of paper length, interested readers may refer to the comprehensive textbook [5] and the rich literature on this topic.

Several image features can be used to detect cut-and-paste forgeries. Cut-and-paste image forgeries consist of replacing a segment of one image with content cut from another. Since the resulting image is completely inauthentic, it is particularly important to identify these types of forgeries. Bicoherence features of an image have been successfully used to identify cut-and-paste forgeries [6]. Additionally, ad-hoc sets of features can be grouped together to create feature sets suitable for detecting cut-and-paste forgeries. Two successful feature sets of this nature are formed by combining moments of an image's wavelet transform coefficients with two dimensional phase congruency features [7] and with Markov transition probability matrices calculated from an image's wavelet transform coefficients [8].

Forensic techniques designed to identify manipulation typically assume that the multimedia file being examined corresponds to a real world signal that may have been subsequently processed. Powerful computer graphics software, however, can allow forgers to create computer generated images that appear photorealistic. As a result, differentiating between computer generated images and those captured by a digital camera is an important forensic task. Forensic techniques that identify computer generated images operate by first gathering a set of features from an image, then by using a machine learning technique to classify the image as real or computer generated. First-order and higher-order statistical features from an image's wavelet transform can be used in this manner to identify computer generated images [9], [10]. Alternatively, a set of geometric features can be used to identify computer generated images consisting of local patch statistics, fractal dimension, and a set of differential geometry features [11]. We will discuss more on this in the context of identifying the type of imaging in Section III-C.

B. DEVICE FINGERPRINTS

Before the emergence of information forensics, fragile and semi-fragile watermarking was proposed as a means of identifying manipulation in multimedia files [12], [13]. These techniques involved inserting an imperceptible, extrinsically generated fingerprint into a multimedia file that would be damaged by subsequent editing. If the recipient of a multimedia file was unable to successfully extract an undamaged version of this fingerprint from the file, they could assume that the file had been manipulated.

Since most multimedia files do not have security measures embedded in them upon file formation, investigators cannot make use of these extrinsic fingerprints. Forensic researchers, however, have identified a number imperceptible traces left in multimedia files by imperfections in an acquisition device's components or as a result of the device's internal processing. As will be discussed in Section III, these traces can be viewed as intrinsic fingerprints that naturally occur within a multimedia file. A number of forensic techniques have been developed to authenticate multimedia files using various different intrinsic fingerprints.

Due to Snell's Law, different wavelengths of light will be refracted different amounts as they pass through a camera's lens. As a result, different color bands of light reflected from the same source will fall incident on slightly different locations on a digital camera's sensor. Inconsistencies in this color distortion, known as chromatic aberration, can be to identify cut-and-paste image forgeries [14]. This is done by modeling the lateral chromatic aberration in an image as a two dimensional linear function. The image's chromatic aberration is then fit to this model by finding the model parameters that minimize the mutual information between each color layer. Next, the chromatic aberration is estimated

on a block by block basis. Forgeries are identified by locating image blocks whose aberration significantly deviates from the global estimate.

Other image manipulation detection techniques make use of the nonlinear manner in which a camera maps pixel illumination values recorded by its sensor into color values. This mapping is known as the camera response function (CRF). One technique for identifying image forgeries using the CRF operates by estimating the CRF across several different edges throughout an image [15]. If inconsistencies exist in certain key properties of the set of estimated CRFs, an image forgery is detected. Additionally, CRF inconsistencies can be used to identify cut-and-paste forgeries [16], [17]. Since different cameras use different CRFs, the CRF will not be consistent throughout a cut-and-paste forgery created from images taken by two different cameras. To identify these inconsistencies, a parametric model can be used to estimate the CRF in locally planar patches on both sides of edges throughout an image. Forged regions will be exposed when the CRFs on both sides of an edge do not match.

Forensic techniques have been developed to detect image forgeries by detecting inconsistencies in the statistical properties of the noise introduced during the imaging process. By using a technique to blindly obtain estimate an image's signal-to-noise ratio (SNR), falsified image regions can be identified by searching for localized inconsistencies in an image's SNR [18]. Inconsistencies in the variance of different blocks within the HH_1 subband of an image's wavelet decomposition can be used to identify falsified image regions [19]. Similarly, an improved noise variance estimation technique can be used to detect image forgeries by identifying localized differences in the noise variance [20], [21]. Additionally, additive noise in previously JPEG compressed color images can be detected by identifying changes to the distribution of an image's luminance histogram after it is subjected to a specific mapping [22].

While inconsistencies in statistical properties of the noise present in an image can be used to identify forgeries, other techniques make use of camera-specific fingerprints noise fingerprints introduced by imperfections in a digital camera's sensor. Forensic techniques to identify forgeries using these fingerprints, known as photo-response non-uniformity (PRNU) fingerprints, will be discussed in detail in Section III. Additionally, Section III will discuss how camera's a color filter array pattern and color interpolation parameters form a set of intrinsic fingerprints, along with how these fingerprints can be used to detect image manipulation.

C. MANIPULATION FINGERPRINTS

1) Fingerprints From Copy-Move Forgeries

In some circumstances, a forger will alter an image by replacing a portion of the image with content copied from elsewhere within the same image. This is often done to hide the presence of an object by covering it with textured elements, such as trees or grass, that are not easily falsifiable. Additionally, it can be used to create duplicates of a significant object within the image. These types of forgeries, known as copy-move forgeries, received some of the earliest attention from the forensics community.

One approach to identifying copy-move forgeries involves segmenting an image into overlapping square blocks, then searching the entire image for duplicates of each block [23]. Blocks are matched on the basis of a set of features extracted from each block. To reduce the computational complexity of the search for duplicate blocks, the DCT coefficients of each block are inserted into the rows of a matrix that is then lexicographically sorted. If the distance between adjacent rows falls below a user defined threshold, the corresponding blocks are considered a match. The threshold can be relaxed to account for the fact that only a portion of some blocks will correspond to a duplicated region. Additional modifications to this basic search approach can be performed to reduce its computations complexity by a factor of four [24].

The performance of this method of identifying copy-move forgeries depends greatly on the features used for block matching. In order to prevent false matches, these features must accurately represent the content within a block. However, for these features to be practically useful, they must be robust to degradations such as noise, compression, and subsequent processing. The quantized discrete cosine transform (DCT) coefficients of each block were proposed as an early set of block matching features [23]. When using these features, the quantization parameters can be adjusted to balance the tradeoff between robustness and matching accuracy. Several subsequent sets of features have been proposed to improve upon the robustness of these features. One approach involves performing principal component analysis on the set of blocks, then retaining the largest principal components of each block as features [25]. Block matching features can be constructed using the mean each color layer along with the ratio of several spatial averages within each block [26]. Additionally, features of each block's Fourier-Mellin transform [27] and blur invariants [28] have been proposed as block matching features.

Recently, an alternate means of identifying copy-move forgeries using scale-invariant feature transform (SIFT) points has been proposed [29]. SIFT is a computer vision algorithm designed to identify and represent certain points as robust local feature descriptors for an image [30]. By identifying correspondences between matching SIFT points within an image, duplicate image regions can be exposed. Unlike techniques that identify copy-move forgeries through block matching, SIFT-based approaches can identify duplicate image regions even if the duplicated region has been subject to scaling, rotation, an affine transform, or illumination adjustment [31], [32].

2) Resampling Fingerprints

Resampling is an image manipulation of particular forensic importance since resampling is performed any time an image is resized, rotated, or subjected to an affine transformation.

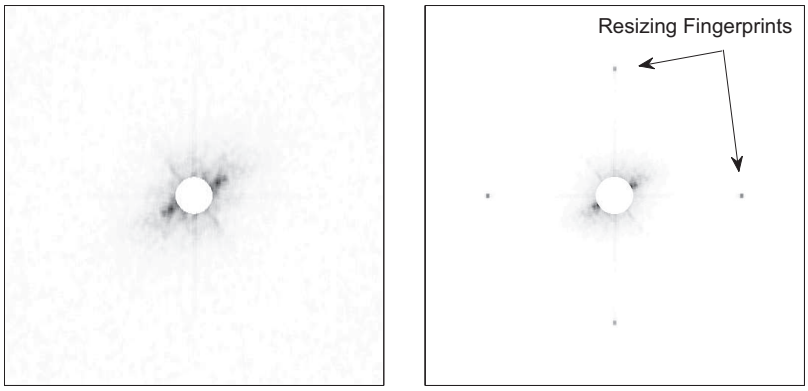


FIGURE 1. **The p-map of an unaltered image (a) along with the p-map of a resized image (b). Note the presence of the spectral peaks corresponding to resizing fingerprints in the p-map of the resized image.**

Though detecting evidence of resampling throughout an entire image does not necessarily imply that the image has been forged, it does indicate that the image has been processed. Furthermore, when creating a cut-and-paste image forgery, resizing or rotation of the object inserted into the background image is often necessary to make the forgery appear visually realistic.

Resampling fingerprints were first identified by noting that resampling will cause many samples of a signal to be correlated with nearby samples [18]. If the signal is resampled by a rational factor, these correlations will be periodic in nature. Resampling fingerprints can be detected by using the expectation maximization (EM) algorithm to jointly estimate a linear prediction filter for each pixel value on the basis of its neighbors along with the probability that a pixel is correlated with its neighbors [33]. This latter quantity, which is referred to as a probability map or p-map, will be periodic if the image has been resampled. Periodicities in the p-map can be identified by inspecting its discrete Fourier transform (DFT) for distinct isolated peaks, which can be seen in Fig. 1.

While this method of detecting resampling is very effective, its reliance on the EM algorithm is computationally expensive. It has recently been demonstrated that the prediction error of an arbitrary linear filter will be periodic if a digital signal has been resampled [34]. As a result, the computational efficiency of this technique can be improved by using a fixed linear filter to calculate an image's p-map. Resampling fingerprints can be automatically detected by comparing the maximum magnitude of the p-map's cumulative periodogram to a threshold [34]. Additionally, the performance of this technique on JPEG compressed images can be improved by suppressing spectral peaks that correspond to 8×8 pixel blocking artifacts and searching for shifted spectral peaks that correspond to resized JPEG blocking artifacts [35].

Alternative means of detecting image resampling have also been developed. One technique detects resampling by identifying periodicities in the average of an image's second derivative along its rows or columns [36]. Additionally, it can be proved that the variance of a resampled signal's derivative is periodic [37]. Using this fact, image resampling can be detected by identifying periodicities in an image's derivative using the Radon transform.

3) Contrast Enhancement Fingerprints

Contrast enhancement is an editing technique often used to alter the illumination conditions within an image. When creating a cut-and-paste forgery, a forger may apply contrast enhancement to falsified regions of an image to ensure that lighting conditions match throughout their forgery. Contrast enhancement operates by applying a nondecreasing nonlinear mapping to the values of a signal. This nonlinear mapping will create contrast enhancement fingerprints in the form of impulsive peaks and gaps introduced into an images pixel value histogram [38]. In locally contractive regions of a contrast enhancement mapping, multiple unaltered pixel values will map to the same contrast enhanced value. This will cause an impulsive spike in the image's pixel value histogram. Similarly, in locally expansive regions of the contrast enhancement mapping will cause adjacent pixel values to be mapped further apart, thus resulting in the absence of values in between from the enhanced image. This will cause a sudden gap in the image's pixel value histogram.

Contrast enhancement can be detected by measuring the strength of these fingerprints in an image's pixel value histogram [22]. These fingerprints can be easily distinguished by examining the Fourier transform of an image's pixel value histogram. Since an unaltered image's pixel value histogram will typically be smooth, the majority of the energy in its Fourier transform will be concentrated in low frequency regions. By contrast, the impulsive nature of contrast enhancement fingerprints will result in a significant amount of energy in high frequency bands. As a result, contrast enhancement can be detected by measuring the amount of high frequency energy introduced into the Fourier transform

of an image's pixel value histogram by these impulsive peaks and gaps [22]. Additionally, the likelihood that a histogram bin corresponds to a contrast enhancement fingerprint can be assessed by using the Laplace distribution to model the difference between each histogram bin and a smoothed version of the histogram. This probabilistic model can be used to jointly obtain a maximum likelihood estimate of the contrast enhancement mapping used to alter an image as well as the image's pixel value histogram before contrast enhancement [39].

4) Median Filtering Fingerprints

Median filtering is another editing operation to receive considerable attention from the forensics community. Due to its nonlinear nature, median filtering is capable of smoothing a signal while preserving its edge content. As a result, median filtering is often used to perform denoising or smoothing in digital images.

One important median filtering fingerprint stems from the observation that median filtering introduces streaking into signals [40]. Streaks correspond to a sequence of adjacent signal observations all taking the same value. As a result, median filtering can be detected in digital images by analyzing statistical properties of the first difference of an image's pixel values. One method of doing this measures median filtering induced streaking by analyzing the ratio of the number of pixel difference whose value is zero to the number of differences whose value is one [41]. Similarly, median filtering can be detected by analyzing the probability that an image's first order pixel difference is zero in textured regions [42].

While these techniques are highly accurate in uncompressed images, their accuracy is greatly diminished if the image has been JPEG compressed. To overcome this, subtractive pixel adjacency matrix (SPAM) features developed to perform steganalysis [43] can be employed to capture properties of an image's conditional pixel difference distributions [41]. A support vector machine (SVM) can then be trained to detect median filtering in JPEG compressed images using these features. Alternatively, a set of ad-hoc features extracted from small image blocks can be used to train a SVM to perform median filtering detection [44].

Recently, statistical properties of an image's median filter residual (MFR) have been proposed as features for robustly detecting median filtering [45]. The MFR is defined as the difference between a median filtered copy of an image and itself. While an image's content will influence statistical properties of its pixel value differences, it has a diminished effect on properties of the MFR. Median filtering can be detected by using an auto-regressive (AR) model to capture statistical properties of an image's MFR, then by training a SVM to identify median filtering using these AR coefficients. This technique has been experimentally shown to achieve important performance gains when identifying median filtering in JPEG compressed images, small image windows, and distinguishing median filtering from other image enhancements.

D. COMPRESSION AND CODING FINGERPRINTS

In order to reduce their size and aid in their storage or transmission over the Internet, the majority of digital multimedia signals undergo some form of coding or lossy compression. Like many signal processing operations, compression and coding leave behind their own distinct fingerprints. As a result, many digital multimedia signals contain some form of compression or coding fingerprints. Due to the ubiquity of these fingerprints, researchers have developed a wide variety of techniques that use compression or coding fingerprints to perform a variety of forensic tasks. These range from identifying forgeries and verifying the authenticity of a multimedia file, to tracing its processing history and determining its origin.

1) Image Compression Fingerprints

Image compression fingerprints are the most widely studied form of compression fingerprints used in forensics. JPEG compression fingerprints play a particularly important role in forensics due to the widespread use of JPEG compression. When a grayscale image undergoes JPEG compression, it is first segmented into a series of nonoverlapping 8×8 pixel blocks. Next, the discrete cosine transform (DCT) of each block is computed, resulting in a set of 64 subbands of DCT coefficients. Each DCT coefficient X is then quantized by dividing it by the entry in a quantization matrix Q that corresponds to its subband, then rounding the resulting value to the nearest integer. As a result, the quantized version Y of a DCT coefficient in the (i, j) subband is given by

$$Y_{i,j} = \text{round}(X_{i,j}/Q_{i,j}). \tag{1}$$

Finally, each quantized DCT coefficient is converted to binary, then reordered into a single bit stream using the zigzag scan order and losslessly encoded.

Color images are compressed in a similar fashion, after they are first transformed from the RGB to the YCbCr color space. Each chrominance layer is then typically downsampled, most commonly by a factor of two in both the horizontal and vertical directions. After this is done, each YCbCr color layer is then compressed as if it were a single grayscale image. Because the human visual system has different sensitivities to luminance and color distortions, different quantization tables are typically used to quantize the luminance and chrominance layers.

When a JPEG is decompressed, each of the steps performed during the encoding process are inverted, with the exception of quantization. Because quantization is a many-to-one mapping, it is non-invertible, therefore dequantization must be performed instead. Each dequantized coefficient $\hat{X}$ is calculated by multiplying its quantized version by the appropriate entry in the quantization matrix, such that

$$\hat{X}_{i,j} = Q_{i,j} Y_{i,j}. \tag{2}$$

As a result, each dequantized DCT coefficient is an integer multiple of the quantization step size. Because each dequantized DCT coefficient is unlikely to correspond to its original

value, the quantization/dequantization process is the main source of image distortion caused by JPEG compression. This distortion, however, results in two forensically significant artifacts used as fingerprints by forensic examiners; DCT coefficient quantization fingerprints and blocking fingerprints.

DCT coefficient quantization fingerprints correspond to the clustering of DCT coefficients around integer multiples of the quantization step size. Since the DCT coefficients of an uncompressed image are continuously distributed, these fingerprints can be easily observed when examining the distribution of each subband of DCT coefficients in an image. They manifest themselves as periodic spikes in the distribution spaced at integer multiples of $Q_{i,j}$. Blocking fingerprints correspond to the image discontinuities that occur across each 8×8 pixel block in JPEG compressed images. These discontinuities are a result of the distortion that is independently introduced into each block by DCT coefficient quantization. Evidence of previous JPEG compression can be identified in an image stored in a lossless format by measuring the strength of each of these fingerprints [46]. Additionally, the quantization table used during JPEG compression can be estimated by obtaining a maximum likelihood estimate of the constant spacing between the histogram peaks in each DCT subband [46].

While JPEG is the most commonly used image compression format, several other lossy compression techniques exist such as wavelet based coders and differential encoders. If a compressed image is later saved in a lossless format, however, it may not be obvious that the image was previously compressed. A forensic framework can be employed to determine the compression history of an image saved in a lossless format [47]. This framework operates by first determining if an image encoder has performed block processing on an image. Block processing is detected and the block size is estimated by examining the average absolute value of the first difference along the row or column direction for periodic spikes [48]. After information about block processing has been gathered, the image is searched for evidence of transform coding. This is done by subjecting the image or image blocks to a set of candidate transforms, fitting the distribution of the resulting transform coefficients to a parametric model, then measuring the distance between the observed coefficient distribution and the model distribution to reveal evidence of quantization [49]. If no evidence of transform coding is identified, the image is searched for evidence of differential encoding by examining its prediction residue for quantization fingerprints.

2) Multiple Compression Fingerprints

Identifying evidence that an image has undergone JPEG compression twice using different quality factors is a well studied and important forensic problem. In addition to providing basic information about an image's processing history, evidence of double JPEG compression reveals that an image may have undergone manipulation since an image must be re-saved after it has been edited. Furthermore, double JPEG compression has important implications for steganalysis [50]. As a result, a number of forensic techniques have been designed to detect double JPEG compression.

As was previously discussed, the values of the DCT coefficients in a JPEG compressed image are integer multiples of the quantization table entry corresponding to their subband. This causes the distribution of each subband of DCT coefficients in a JPEG compressed image to resemble a smoothly decaying periodic impulse train. If the image is compressed again using a different quantization table, the coefficients of many DCT subbands will be requantized using a different quantization step size [50], [51]. The resulting mismatch in quantization intervals will cause either periodic spikes or zeros in the amplitude of these impulses depending on the relationship between the two quantization step sizes. These periodic artifacts correspond to fingerprints left by double JPEG compression.

Several techniques exist to detect double JPEG compression fingerprints throughout an image. A neural network can be trained to both detect double JPEG compression and estimate the quantization matrix used during the first compression on the basis of an image's DCT coefficient histograms [50]. The performance of this approach can be improved using a SVM trained on the histograms of several DCT subbands [52]. Alternatively, the Fourier transform of the histogram of each subband of DCT coefficients can be inspected for spikes that correspond to the periodic fingerprints of double JPEG compression [51]. Additionally, double JPEG compression using the same quality factor can be detected by analyzing the number of DCT coefficients whose values change after subsequent JPEG compression is applied to an image [53].

The distribution of the first digit of an image's DCT coefficients can also be used to detect double JPEG compression [54]. In a single JPEG compressed image, it has been observed that this distribution can be modeled using a modified version of Benford's law. Since the first digit distribution of DCT coefficients from a double JPEG compressed image will not fit this model, double JPEG compression can be detected by examining how well the first digit distribution of an image's DCT coefficients fits a modified Benford distribution. The performance of this technique can be improved by examining the first digit distribution of each subband of DCT coefficients independently rather than analyzing the entire set at once [55].

The detection of localized evidence of double JPEG compression is of even greater forensic significance because it can expose cut-and-paste forgeries. Localized double JPEG compression fingerprints take two different forms. The first corresponds to mismatches in properties of an image's DCT coefficient histograms. These mismatches can arise due to a number of reasons. If only one of the two images used to create a cut-and paste forgery was previously JPEG compressed, then periodic double JPEG compression fingerprints will only occur in regions corresponding the image that was

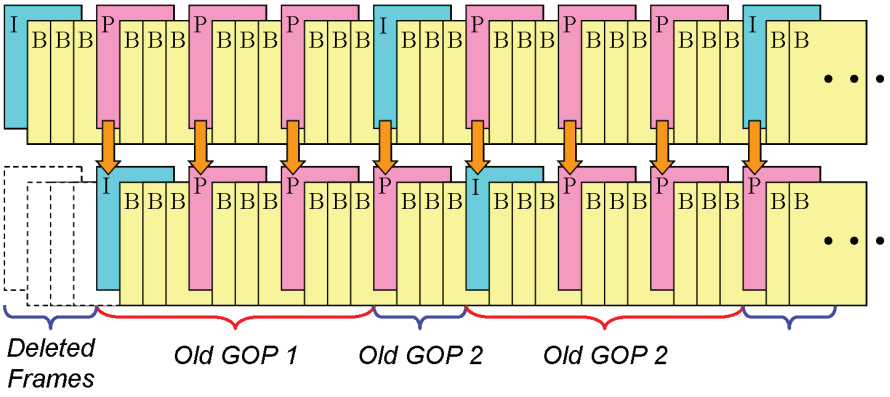


FIGURE 2. Illustration of the effects of frame deletion on a video frame sequence. The original video sequence is shown along the top of this figure and the altered video sequence is shown along the bottom. Each GOP in the altered video contains frames from two different GOPs in the unaltered video sequence.

previously compressed. Similarly, if the blocking grids of the background image and pasted image do not align when the forgery is recompressed, the pasted region will not contain double JPEG compression fingerprints in the DCT domain. Additionally, if the two images were previously compressed using different quantization tables, the periods of the DCT domain double JPEG fingerprints will be different in each image region. A number of forensic techniques have been designed to identify cut-and-paste image forgeries by detecting localized inconsistencies in DCT domain double JPEG compression fingerprints [56]–[61].

The second type of localized double JPEG compression fingerprint corresponds to irregularities in the forgery's blocking artifacts. If the 8×8 pixel blocking grid of the falsified portion of the image does not match the blocking grid of the background, this region will contain two sets of blocking artifacts. If double JPEG compression occurs when creating a forgery, it is likely that these fingerprints will be present because they will occur in 63 of the 64 possible blocking grid alignments. As a result, many forensic techniques have been designed to detect these blocking grid misalignment fingerprints [62]–[65].

3) Video Compression Fingerprints

Compression fingerprints play an important role in video forensics. Virtually all digital videos undergo compression because the size of uncompressed video files make it impractical to store or transmit them. As a result, forensic investigators can reliably expect compression fingerprints to be present in digital videos. These compression fingerprints fall into two basic categories: spatial and temporal. Spatial fingerprints are isolated to a single video frame and resemble image compression fingerprints. Temporal fingerprints, however, are spread throughout the entire video. A number of forensic techniques have been designed to exploit the spatial and temporal fingerprints left by video compression.

While a variety of video compression standards have been proposed, the majority operate in the same basic manner. Rather than processing the entire video at once, an encoder will divide the sequence of video frames into smaller chunks. Each chunk, known as a group of pictures (GOP), begins with an intra-frame (I-frame). I-frames are independently encoded using a process similar to JPEG compression. The remainder of the GOP consists of predicted-frames (P-frames) and bidirectional-frames (B-frames).

P-frames are predicted from I-frames or P-frames that occur before them in the video sequence using a process known as motion compensation. During motion compensation, the frame to be encoded is divided into macroblocks. A prediction of the frame is created by searching the previous frame for the block that best matches each macroblock. The prediction error is stored by the encoder using a process similar to JPEG compression, along with the motion vectors corresponding to the row and column displacements between each matching macroblock. B-frames are encoded in a similar manner, however for these frames, macroblocks can be predicted from I- or P-frames that precede or follow each B-frame.

Since I-frames are encoded using a process similar to JPEG compression, I-frames contain spatial fingerprints that are nearly identical to those left by JPEG compression. As a result, double MPEG compression can be detected by searching for double quantization fingerprints in the DCT coefficients of a video's I-frames. This can be done by first using a probabilistic model to calculate the probability that the DCT coefficients in each macroblock in an I-frame have been quantized twice, then by identifying macroblocks that are likely inauthentic [66]. Additionally, spatial double compression fingerprints can be detected in I-frames using techniques originally designed to identify double JPEG compression.

If a sequence of frames are deleted from a digital video, temporally distributed fingerprints will be introduced into the sequence of the video's P-frame prediction errors [67]. Frame deletion will cause a shift in the sequence of video frames, resulting in the formation of a new set of GOPs when the video is recompressed. When this happens, some P-frames

within a GOP will be predicted from I-frames or P-frames that previously belonged to a different GOP. An example of this can be seen in Fig. 2. This will result in a sudden spike in the prediction error corresponding to frame deletion fingerprints.

Two different models of frame deletion fingerprints have been built depending on the GOP structure used by the video encoder [68]. If a fixed GOP structure is used during compression, the spikes in the P-frame prediction error sequence will occur periodically. As a result, these fingerprints can be detected by searching for spikes in the Fourier transform of the P-frame prediction error sequence. If, however, variable GOP lengths are used, this sequence will be aperiodic. These fingerprints can be detected by measuring the energy of the difference between the P-frame prediction error sequence and a smoothed version of itself.

E. PHYSICAL INCONSISTENCIES

All of the manipulation detection techniques described thus far in this section operate by making use of some type of forensic fingerprint. One final family of multimedia authentication techniques exists that does not use any form of fingerprints. Instead, these techniques detect falsified multimedia files by using physics-based models to identify inconsistencies within the content of multimedia files. This is often accomplished by making use of concepts developed by computer vision researchers.

One set of techniques identifies image forgeries by checking for lighting inconsistencies in a scene. In early work, this was accomplished this by obtaining a two dimensional estimate of the direction of the light source illuminating an object [69]. This technique exploits the fact that by estimating lighting angle along an object's occluding boundary, two of the three dimensions that define the location of the light source can be estimated from a single image. Image forgeries can be identified by calculating the lighting angle for several objects in an image and checking to ensure that they all correspond to a common source.

While the previous technique assumes that a scene is illuminated by a single light source, this is often not the case. To address this, a more advanced method exists to estimate the locations of multiple light sources in a scene [70]. Alternatively, a three dimensional estimate of the direction of the lighting source can be obtained using specular reflections in the human eye [71]. This is accomplished by modeling the cornea as a spherical surface. By doing this, the surface normal is known on all points on the cornea and the three dimensional lighting angle can be resolved.

Video forgeries can be exposed by identifying projectiles in videos whose path of motion violates physical laws [72]. This is done by observing that a projectile's motion through three dimensional space should take a parabolic path. To verify that the motion of a projectile in the video has not been falsified, the path of the object through the video can be tracked and the trajectory of the object is estimated. Video forgeries can then be identified by testing to see if this trajectory significantly deviates from the parabolic motion model.

III. DEVICE FORENSICS

With the advancement of digital imaging technologies, we have seen a growth in the popularity of digital cameras and images. Tracing where an image is from and how it was generated is an important step in ensuring the security and trustworthiness of such digital data. As a result, the need for conducting forensic analysis of an acquisition device is becoming common in today's crime scene investigation. Similar trends can be seen from fighting against terrorists and protecting homeland security, to mitigating business and industrial espionage. It is crucial to ensure the trustworthiness of a digital image before it can be used as forensic clues or as evidence in court.

An easy option for imaging device linkage is through the metadata associated with an image file, such as JPEG file headers. Such metadata, however, are not always reliable, especially in an adversarial environment where the metadata may be altered to mislead forensic analysis. Instead, individual processing modules in the imaging pipeline, either hardware or software, leave in the final output image certain "intrinsic traces" that potentially carry useful information about the imaging history. Compared to metadata, such intrinsic traces are much more difficult to modify, and therefore may serve as a more robust tool for imaging device linkage. This methodological framework, referred to as *component forensics*, is designed to collect evidence of what techniques and parameters are employed in various hardware and software modules used to produce a digital visual output [73].

More specifically, component forensic techniques look for unique inherent device traces that can be used to differentiate between imaging mechanisms, device models, or device units. The traces to identify one aspect of an image's originating device, such as individual units, may be different from such other aspects as brand or model. In order to differentiate between individual device units, especially those coming from the same manufacturer and model line, it is necessary to exploit features that capture individuality. Variations in a device's physical or analog sensing modules are potential sources for revealing individuality. For example, imperfections in the camera manufacturing process lead to slight variations among photon sensors in their photon-electron conversion ability. As a result, each individual camera unit has its own variation pattern.

Beyond these individual variations, it is necessary to answer such questions as what is common among all iPhone 5 cameras but different from Samsung, and what is common among all pictures generated by digital cameras but different from those that are scanned. The evolution and coexistence of analog and digital processing suggest two possible sources of intrinsic traces to examine: certain ensemble characteristics of physical/analog sensors between two manufacturers or model lines, and characteristics from in-camera software processing that is the often the same across units from the same brand or model but different between different brands or models. For example, noise strength and other statistical

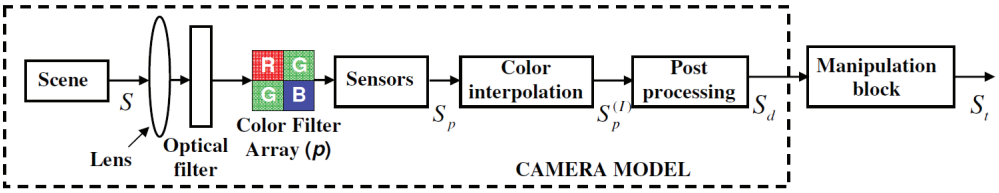


FIGURE 3. Imaging process in digital cameras.

characteristics (instead of the specific noise patterns in the above case) may be different between manufacturers because of the different technologies or processes used, and may also reflect the differences in imaging mechanisms. Another example that we shall discuss next is color processing in today's digital cameras. Because different camera brands and models employ their own color-processing methods and parameters, color features can be potentially useful traces to trace down to brand or model.

A. EXPLORING COLOR PROCESSING TRACES

As mentioned above, component forensics was proposed as a framework to collect evidence of which techniques and parameters are employed in various hardware and software modules used to produce a digital visual output. Here, we use images captured by digital cameras as an example to present the framework and approaches to component forensics.

The first step of component forensics for a digital camera is to understand the process of how a digital image is generated. As shown in the dashed box of the schematic diagram in Fig. 3, when a scene is captured by a digital image or video camera, light from the scene first passes through the camera's lens and optical filters and is then recorded by CCD sensor arrays. Few consumer-level color cameras directly acquire full-resolution information for all three primary colors (usually red, green, and blue). This is not only because of the high cost in producing full-resolution sensors for all three colors, but also due to the substantial difficulty involved in perfectly matching the corresponding pixels and aligning the three color planes together. For these reasons, most digital cameras and camcorders use a color filter array (CFA) to sample the real-world scene [75]. The CFA consists of an array of color sensors, and is typically a pattern tiled with 2×2 cells of R/G/B sensors. Each sensor captures the corresponding color of the real-world scene at its pixel location. The remaining color values of the pixel are interpolated from other directly observed values in its neighborhood [76]. For example, in the CFA array shown in Fig. 3, we have one red color value, one blue value, and two green values for every four pixels in a 2×2 neighborhood, resulting in a set of three partially sampled color layers. The unobserved values in each color layer are then filled in using an interpolation algorithm. After interpolation, the three color layers corresponding to the red, green and blue components go though a post-processing stage. Color corrections such as white balancing are done in this stage, and the image or image sequence may undergo lossy compression (e.g. via JPEG or MPEG) to reduce storage space.

The performance of color interpolation directly affects the quality of the image captured by a camera, as artifacts in an interpolated picture will be propagated and may be amplified by subsequent processing modules [77]. As such, color interpolation has become one of the major components that various camera manufacturers intensively compete on. Different manufacturers often have different variations of schemes. From a component-forensics perspective, color interpolation is a promising component to provide evidence that can differentiate images taken by different cameras. There are several commonly used algorithms for color interpolation in the literature, such as bilinear, bicubic, smooth hue, and gradient based [77], [78]. These algorithms can be broadly classified into two categories, non-adaptive and adaptive algorithms, based on their ability to adapt the interpolation algorithm according to local image characteristics [76]. For example, edge adaptive algorithms tries to perform interpolation along edge directions instead of across edges in order to retain sharpness in the interpolated image. The CFA interpolated image then goes through a post-processing stage where white balancing and color correction are done to remove unrealistic color casts in the image.

To cope with the dependency among multiple interconnected components inside a camera, a set of robust and non-intrusive algorithms can be developed that make inference from output images to jointly estimate the parameters in these multiple components. Taking color interpolation as an example, both the CFA pattern and interpolation coefficients are unknown and closely related. As a result, they must be estimated together. Since a camera may employ different interpolation filters for different types of regions, it is desirable to first classify each image pixel into one of several groups, for example, with the local neighborhood having significant horizontal edges, vertical edges, or being mostly smooth. It is also necessary to establish a search space of CFA patterns and use knowledge of common practices in sensor design to keep the search space as small as possible. For each CFA pattern p in the search space, the set of pixels in the interpolated image $S_p^{(i)}$ obtained directly from CCD sensors and those obtained by interpolation are first identified. After this, a set of linear equations can be established relating the values of all the interpolated pixels to directly observed ones. To overcome noisy values in these equations that are due to

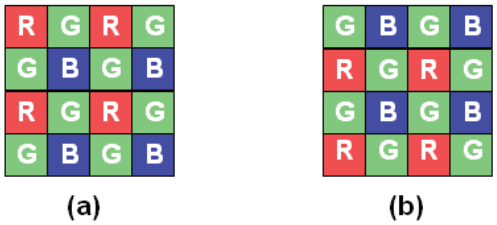


FIGURE 4. CFA patterns for (a) Canon EOS Digital Rebel and (b) Fujifilm S3000 cameras.

TABLE 1. Confusion matrix of classification results on camera brands.

(with * denoting insignificant values below 3%)

	(C)	(N)	(S)	(O)	(M)	(Cs)	(F)	(E)
Canon (C)	98%	*	*	*	*	*	*	*
Nikon (N)	6%	85%	5%	*	*	*	*	*
Sony (S)	*	*	93%	*	*	*	*	*
Olympus (O)	6%	6%	*	85%	*	*	*	*
Minolta (M)	*	*	4%	*	91%	*	*	*
Casio (Cs)	*	*	*	5%	*	91%	*	*
Fuji (F)	*	*	*	*	*	*	95%	*
Epson (E)	*	*	*	*	*	*	*	100%

operations following interpolation (such as JPEG compression), a least squares method is used to estimate the interpolation coefficients for each type of region. Finally, these estimated coefficients are used to perform color interpolation and the resulting image can be is compared with the the output image S_d from the camera. The CFA pattern $\hat{p}$ that gives the lowest interpolation error and its corresponding coefficient estimates are chosen as the final estimation results.

To demonstrate this joint estimation approach, we used two cameras that offer raw sensor output containing the ground-truth information of CFA pattern. Using JPEG images obtained from a Canon EOS Digital Rebel camera as input, the joint estimation algorithm reveals that the CFA patterns shown in Fig. 4(a) minimizes the interpolation error; similarly, the estimated CFA pattern for Fujifilm S3000 camera is in shown Fig. 4(b). These results agree perfectly with the ground-truth supplied by the cameras' raw sensor output file. The estimated interpolation coefficients corresponding to the CFA pattern can be further fed into SVM classifiers to infer what kind of interpolation algorithm was employed by the camera (e.g. bilinear, bicubic, edge-directed interpolation, etc.) [74]. With the use of the least square formulation in solving for interpolation parameters, it has been shown that the obtained parameters have very good resilience against compression and other post-processing in identifying the type of interpolation algorithms used to produce a camera output image [79].

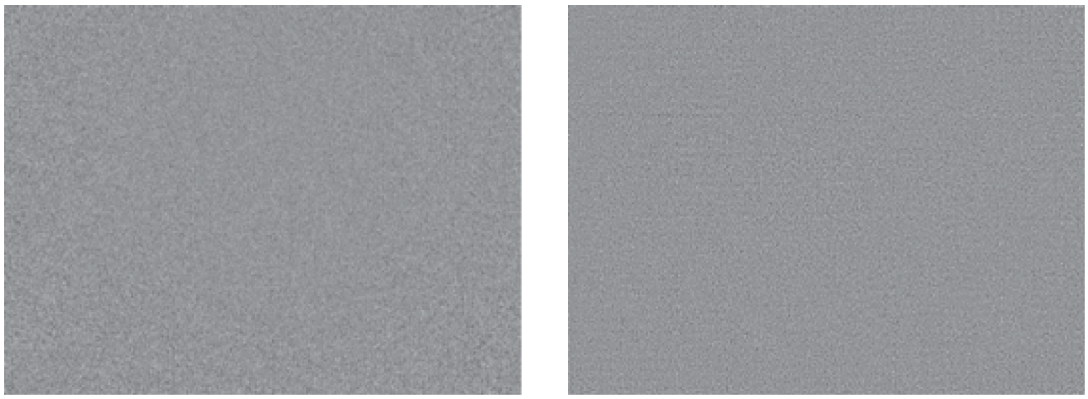
This type of camera component analysis can provide a number of distinguishing features to help identify which camera has been used to capture a given image. For example, the estimated algorithm and parameters from color interpolation can be used as features to build a classifier for camera identification. As a simple demonstration, we considered 16 different cameras and collected 40 images taken by each camera in an uncontrolled environment. The images taken by different cameras generally have different scene content and are compressed under default JPEG quality factors as specified by the cameras. The color interpolation coefficients were estimated using the algorithm outlined above and were used to perform classification. We considered each manufacturer as one class (which may consist of different models of the same manufacturer) and built an 8-class classifier. We randomly chose 25 images to train an SVM and then test on the remaining 15 images. This process was repeated 200 times and the average classification results are computed. The results are put together in a confusion matrix shown in Table 1. Here, the $(i, j)^{th}$ element in the confusion matrix gives the probability of being classified as camera brand$-j$ when the picture actually comes from camera brand$-i$. We can see that main-diagonal elements has an average value of 92.25% for 8 camera brands.

The classification performance can be further enhanced by exploiting more detailed models to handle different gradient regions [80], and account for more sophisticated color interpolation techniques [81]. It is also possible to use the same set of features to build a classifier to differentiate between specific model lines. For example, the likely upgrade in software processing across model families from the same manufacturer could be used to distinguish between an iPhone 3 and an iPhone 5. Along this line, the similarities and differences between the component forensic features may shine light on how the technologies have evolved. In case of close similarities observed between different manufacturers, forensic studies provide an interesting and unique lens into technology cooperation in industry (such as by OEM from a common supplier or by technology licensing if legitimately done), or serve as a smoking gun for potential patent infringement [73].

Although explicit estimation of CFA interpolation parameters in the component forensic framework provides an encouraging accuracy under uncontrolled settings and a scalability to a large number of brands/models, alternative approaches to camera identification can be found in the literature employing features that bypass such explicit estimations, possibly at a somewhat reduced computational complexity. In [82], the weighting coefficients from the EM algorithm in [78] and the peak location and magnitudes of the frequency spectrum of the probability map are used as features for classification [82]. When images captured from two cameras under controlled input conditions along with randomly acquired images from the Internet for a third camera were used in a set of experiments, the authors reported accuracies close to 84% on three brands [82]. Further improvements to this algorithm were made in [83] by separately considering smooth and non-smooth regions in the image to obtain accuracies close to 96% for three camera brands.

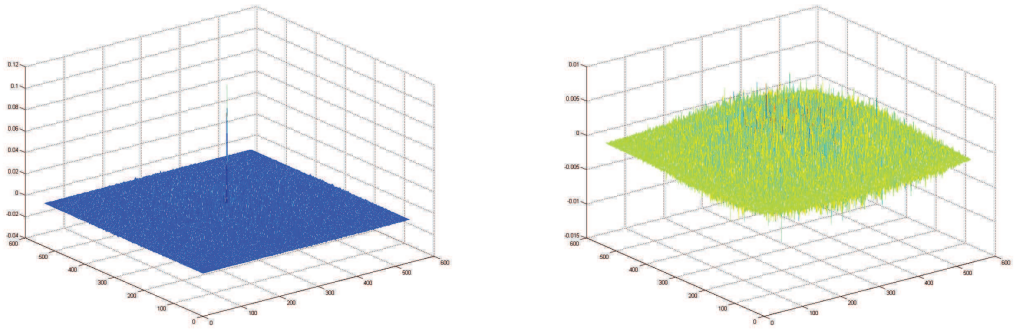
B. LINKING TO INDIVIDUAL DEVICE UNITS

As discussed in the opening of this section, in order to differentiate between individual device units, especially those



(a) PRNU pattern of an Apple iPhone 3G. (b) PRNU pattern of a Samsung i760.

FIGURE 5. PRNU examples from two cellphone cameras. These are best viewed using the electronic version.



(a) Detection showing a match between two PRNUs. (b) Detection showing unmatch between PRNUs.

FIGURE 6. PRNU matching and unmatching examples. (a) Detection showing a match between two PRNUs. (b) Detection showing unmatch between PRNUs.

coming from the same manufacturer and model line (such as two different iPhone 5 units), it is necessary to exploit features that capture individuality. Instances from the variations of a device's physical/analog sensing modules are potential sources reflecting the individuality. Intrinsic features that are difficult to manipulate or duplicate, often known as *physically unclonable features (PUF)*, can be particularly attractive for this forensic task. For images taken by cameras, linking an image to a particular camera unit can be achieved by exploiting the "noise" hidden in the image.

Digital cameras that use CCD or CMOS sensors contain a number of sources of noise that affect their output digital images. One of the main noise sources is due to manufacturing variations or imperfections when the camera was produced. These can cause each photonic sensor element in a camera to have slight variations in their sensitivity to incident light. As a result, different pixels may register different values even under the same light intensity. Such unevenness of light sensitivity is referred to as *Photo Response Non-Uniformity (PRNU)* [84]. The unique PRNU pattern of each camera sensor can thus serve as a fingerprint to characterize an individual camera's sensor array. Sample PRNU noise patterns for two cameras are shown in Fig. 5, wherein brighter pixels correspond to pixels that are more light-sensitive.

As the effect of the PRNU is generally proportional to the intensity of the incident light, the amount of PRNU at each pixel can be modeled as a multiplicative noise-like signal [85] given by $I = I^{(0)} + I^{(0)}K + \theta$. Here, I is the pixel intensity recorded by the given pixel of the camera sensor, $I^{(0)}$ is the ideal sensor output in the absence of noise, K represents the PRNU noise at the pixel per unit amount light, and θ accounts for the combination of other noise sources. Assembling the estimate of K from each pixel gives the overall PRNU pattern for the camera sensor. The PRNU pattern can be obtained by first applying denoising to extract the noise components from each of N training images that are taken by the camera of interest. After this, an estimate of the PRNU is computed using the maximum likelihood estimation method. For a given pixel in the camera's PRNU pattern, this most probable estimate of the PRNU is shown to be:

$$K = \frac{\sum_{i=1}^{N} W_i I_i}{\sum_{i=1}^{N} (I_i)^2}$$

where W_i corresponds to the noise residual obtained by denoising and I_i the light intensity in the i^{th} image [85].

Similarly, to determine whether an image under question (the testing image) has been taken by a specific camera, the noise pattern can first be extracted from the testing image. The similarity can then be compared between this noise pattern and a camera's PRNU pattern. A significant relative peak in the normalized cross correlation suggests the image is highly likely taken by the camera in question. Fig. 6 shows a typical correlation plot between two matched patterns and between unmatched patterns, respectively. Strategic weighting techniques can be further employed during the detection to deal with different sources of inaccuracy in PRNU noise estimation [86].

Image-based PRNU has been extended to identifying specific digital camcorders from videos taken by these devices [87], and the influence of video content on identification performance has been examined in [88]. An increasingly popular way to generate videos is by the more ubiquitous cell-phone cameras. As cell-phones generally apply stronger compression to video captured, this challenge needs to be properly addressed. Studies in [89] found that different video frame types (I- and P- in the MPEG family) have different levels of reliability for PRNU estimation. By reordering and applying proper weights to frames in a video according to their reliability, more accurate identification of the source camera can be achieved by processing a small fraction of frames.

C. IDENTIFYING IMAGING TYPE

In a practical forensic investigation scenario, there are many potential unknowns, and it is desirable to develop conjectures and guide investigations into the origin of an image in a logical fashion. For example, linking an image to a specific camera unit often requires having a ''suspect'' device available to test with. Without additional information, this can be a task of finding a needle in a haystack. Using color processing and other related features can help verify the trustworthiness of the camera brand/model information at the image header or identify the correct brand of camera, but how do we know an image was taken by a camera in the first place? It will be helpful if an investigator can first determine what imaging mechanism has been used to generate the image, as this can significantly narrow down the search range for the next step of the investigation, or help discover discrepancies, such as identifying that an photo-realistic image was in fact generated by computer graphics [11].

As was discussed briefly in the previous section, a camera response function (CRF) relates image irradiance at the image plane to the measured intensity values. The work in [11] explored physical insights from imaging and differential geometry characteristics commonly seen in computer graphic rendering. It was discovered there that the CRF revealed from gradient patches of an image can be substantially different from those rendered by computer graphics. Polygonal models that are commonly used in computer graphic rendering can result in unusually sharp edges and sharp corners in the image intensity function as compared to natural photographic images. Since computer graphic rendering often assumes color independence in the object surface model, which is generally not true for real-world object, the difference in the RGB correlation for photographic and computer graphic images can be exploited using the surface Laplacian. Further studies on reliable estimation of CRF were made in [90].

In addition to the camera response function and differential geometric characteristics, noise properties have been shown to be effective discriminating features as well. We have seen that the PRNU pattern related to noise introduced in the imaging process can be used to differentiate between camera units. Although multiple noise sources may contribute as a whole to the pixel value variation and these noise sources appear highly random in nature, their statistical characteristics may be relatively stable with respect to the same imaging or manufacturing mechanism. While the imaging devices may apply post-processing to reduce noise or for other purposes, some statistical properties of noise remain. In [91], [92] and follow-up developments of [93], [94], noise characteristics were explored from several angles: by employing various denoising algorithms to extract noise and examine the noise strength and statistical variations; by analyzing the statistical properties of the Wavelet coefficients and filterbank outputs of images; and by applying prediction using neighborhood pixels and examining the statistical attributes of the prediction residues. Features collected through these means can be used to train a classifier, for example, a SVM, which may possibly be preceded by dimensionality reduction techniques as principal component analysis (PCA) or linear discriminative analysis (LDA), so that the most likely imaging type can be determined for an image under question.

Combining such forensic features as noise statistics [95] and noise plus color interpolation features [96] with machine learning techniques, algorithms have been developed to determine the most likely type of imaging techniques or devices that was used to generate an image under question. For example, to determine if an image is from a scanner, or computer graphic rendering, or a digital camera, or a low-end cell-phone camera.

D. MANIPULATION DETECTION USING DEVICE FINGERPRINTS

As we first discussed in Section II-B, not only can device fingerprints identify the camera used to capture a digital image, but they can also be used to expose image manipulation. This holds true for PRNU fingerprints as well as a camera's CFA pattern and color interpolation parameters. When a forger creates a falsified image through cut-and-paste forgery, this often involves splicing together parts of pictures captured using different cameras. It is likely that these cameras may employ a different set of algorithms, parameters, or characteristics for their internal components. Examining the inconsistencies of device traces as well as the absence of anticipated traces or

FIGURE 7. Example of tampering detection based on inconsistencies of CFA estimates between regions: (a) a forgery image; (b) editing map showing regions in the forgery image obtained from the two cameras; (c) estimation result of acquisition camera for each block of (a). Black: Sony P72; white: Canon S410; grey: regions with low classification confidence (mostly smooth regions that do not reveal significantly different CFA traces among manufacturers).

the presence of additional traces are common methodologies to detect tampering.

Since the overwhelming majority of consumer level digital cameras employ CFAs in their image processing pipeline, unaltered camera images should contain CFA interpolation fingerprints. CFA interpolation fingerprints may be absent, however, from manipulated images. To detect these fingerprints, an expectation maximization (EM) algorithm can be used to jointly obtain a linear estimate of the CFA interpolation filter's coefficients as well as the probability that a pixel is correlated with its neighbors due to CFA interpolation [78]. Forgeries can be detected by identifying image segments that do not contain CFA interpolation fingerprints. Another approach builds on the component forensic framework reviewed in Section III-A. This approach operates by obtaining regional estimates of the CFA interpolation parameters and classifies the parameters according to the best matching camera model learned from an offline training phase. It then examines the consistencies of the classification results between region. Forgeries can be exposed by identifying mismatches in the origin of different image regions. An example of forensic results obtained in this manner is shown in Fig. 7.

Other techniques with lower computational complexity can be found in the literature to detect the presence of CFA interpolation fingerprints without requiring the estimation of the CFA interpolation filter. One method involves searching for periodicities in the variance of the output of a high-pass filter that has been averaged along one of the diagonal directions [97]. Certain forms of these periodicities will reflect the CFA interpolation fingerprints. Other techniques are designed to specifically detect the localized presence or absence of CFA interpolation fingerprints. One such method operates by analyzing the uniformity of the error induced by reapplying CFA interpolation, while another similar method operates by comparing the power of statistical noise present in interpolated pixels to the noise power in directly observed pixels [98].

Additionally, CFA fingerprints can also be used to identify editing operations that have been globally applied to an image [99]. This is done by approximating an editing operation as a linear filter, then using an iterative process to obtain an estimate of this filter through blind deconvolution. The iterative process works by first obtaining an estimate of the original image by performing a ''deconvolution'' in the frequency domain using the current estimate of the manipulation filter. Next, properties of the color interpolation coefficients are imposed as a set of constraints on the in the pixel domain and used to refine the estimate of the manipulation filter. This process is iterated until a stable estimate of the tampering filter is obtained.

Once an estimate of the manipulation filter has been obtained, image tampering is detected by calculating a similarity score between the estimate of the tampering filter and estimates gathered from unaltered images (which, if if round-off errors are omitted, should correspond to an identity transform with a flat spectral response). This technique is able to detect manipulation in the form of a number of global image processing operations such as linear smoothing, median filtering, resizing, rotation, histogram equalization, JPEG compression, and the addition of noise [100]. Examples of the estimated manipulation filters under various operations are shown in Fig. 8. Since the estimate of the manipulation filter reflects the linear-shift-invariant component of the ''empirical frequency response'' of the operation that the image has undergone, by examining the characteristics of this response shared among the same type of operations (such as JPEG compression and filtering), the type of manipulation can be further identified [101]. This blind-deconvolution approach can also be used to perform universal steganalysis to determine whether an expected camera output image has additional data hidden in it, and can detect both quantization based and spread-spectrum based steganography [100].

PRNU fingerprints constitute another important form of device fingerprints that can be used to identify falsified images. As was discussed in Section III-B, imperfections in a digital camera's sensor cause the sensor's response to light to vary from pixel to pixel. The resulting PRNU noise pattern left in a digital image or video frame is unique to that device. If the PRNU of the camera that captured a digital image is known, it can be used to identify falsified regions in the image [102]. This is done by first computing the correlation between the PRNU estimated from an image and the PRNU of its camera of origin over a sliding window. Inauthentic image

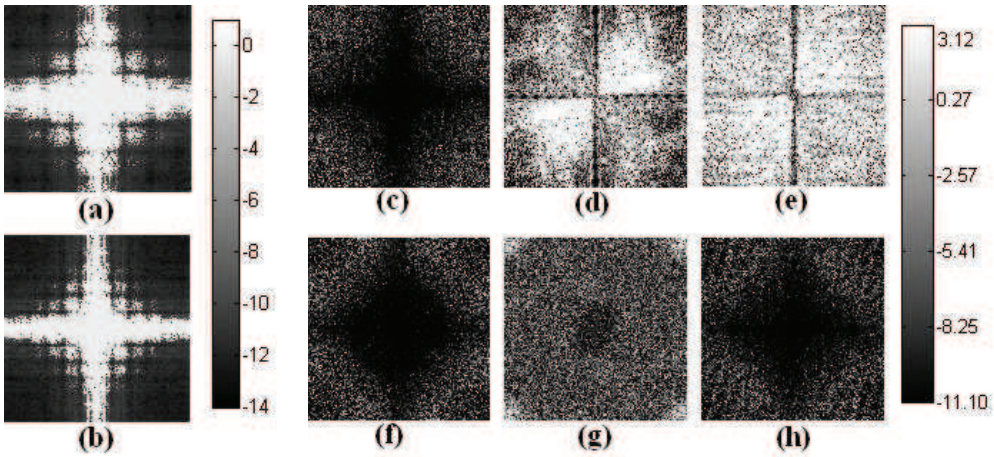


FIGURE 8. **Frequency response of the manipulation filter for camera outputs that are manipulated by (a) 7 × 7 averaging filter, (b) 11 × 11 averaging filter, (c) 7 × 7 median filter, (d) 20 degrees rotation, (e) 70% resampling, (f) 130% resampling, (g) noise addition with PSNR 20dB, and (h) histogram equalization. The frequency response is shown in the log scale and shifted so that the DC components are in the center. (Source: [100])**

regions are detected if the correlation score falls below a certain threshold. To determine the optimal threshold, the correlation score can be modeled using a generalized Gaussian distribution with different distribution parameters depending on whether the block has been tampered or not. A Neyman-Pearson rule for identifying falsified image regions can then be designed to determine the optimal detection threshold for a desired false alarm rate [103]. Changes in noise statistics can also be explored to identify postprocessing operations applied to scanned images [94], [104].

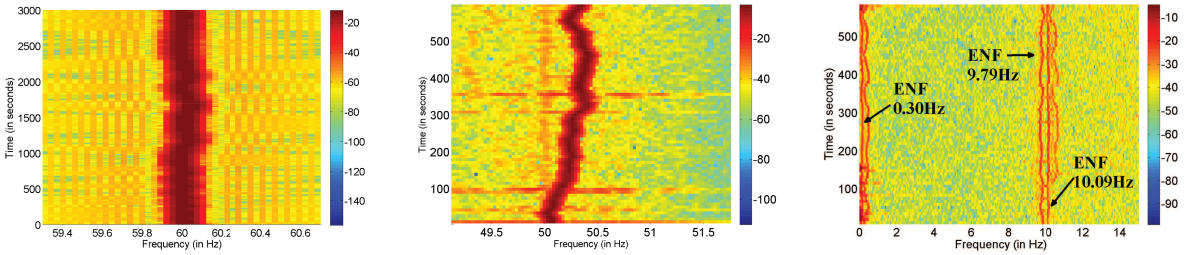
IV. EMERGING FORENSIC USE OF ENVIRONMENTAL SIGNATURES

In this section, we discuss an emerging class of forensic techniques that exploit signatures from a sensing environment. To motivate these techniques, consider the propaganda videos that were periodically released by Osama bin Laden. When faced with these videos and videos like them, counter-terrorism analysts and officials worldwide will seek the answers to many information forensic questions: given a video in question, when and where was it shot? Was the sound track captured together at the same time/location as the visuals or was it superimposed later? Similar questions about the time, location, and the integrity of multimedia and other sensor recordings are also important to provide security and trust in law enforcement, journalism, infrastructure monitoring, smart grid/energy management, and other commercial operations.

Although an increasing number of devices are now equipped with GPS capabilities and many offer user-configurable time stamps, such time and location information is not always available or trustworthy. The adversarial situations that many forensic applications must work in prompt a need for examining complementary types of evidence that can reveal the time and location in which media and sensor information were acquired. Here, we review emerging techniques that hold an encouraging potential to address these questions by exploiting novel fingerprints from the environment. Such environmental fingerprints become naturally "embedded" into videos or other common types of sensor signals at the time of recording. They may carry time and location information, and may facilitate verification of the integrity of the sensed data. One important example of environmental fingerprints arises from the power grid in the form of small random-like fluctuations of the electricity frequency.

The electric network frequency (ENF) is the supply frequency of power distribution networks in a power grid. Its nominal value is 60 Hz in the United States, and 50 Hz in Europe and a large part of Asia. Digital devices, such as audio recorders, CCTV recorders, and video cameras, that are connected to the power mains or are battery powered and located near power sources, pick up the ENF because of influences from electro-magnetic (EM) fields generated by power sources. Although the ENF is closely regulated to stay at the nominal value, its value actually fluctuates from the nominal because of time-varying loads on the power grid and the dynamic control process to match electricity supplies with demands [105]. The main deviations from the nominal value are consistent across the same power grid because of its interconnection nature. There are three major grids in North America; one for the east, one for the west, and one in Texas [106]. The ENF fluctuations in the United States vary randomly and continuously between 59.90 Hz and 60.10 Hz [107]. Similarly in Europe, these fluctuations are typically between 49.90 Hz and 50.10 Hz [108]. The power grid in Asia is less tightly regulated, leading to noticeable ripples observed in China and significant drift as much as 0.5Hz in India [109], as shown in Fig. 9.

It has been demonstrated that the ENF can be "heard" in sound recordings by electricity powered and many battery powered devices [107], [108]. Additionally, it has recently been shown in a proof-of-concept demonstration that the ENF is detectable from visual recordings with indoor



(a) from power line in U.S. (b) from power line in India (c) from video signal in China

FIGURE 9. **Spectrogram showing ENF signals observed in different countries and modalities.**

lighting. The low frame rate and low signal-to-noise ratio in video, however, impose challenges that call for advanced signal processing theory and techniques to solve this problem [109].

What can ENF do for us? ENF signals can be continuously recorded from a power outlet or through such wire-free means as audio/visual sensing to serve as a reference for the respective power grid. Since the random fluctuation of the ENF vary over time, the ENF can be extracted from a sensor recording of interest via a variety of frequency estimation methods [108]–[111] and aligned with a reference ENF signal to determine or verify the recording time. The forensic use of ENF to authenticate audio recordings was first demonstrated in Europe [108], and has been incorporated into the forensic practices by some European countries [112].

With ENF reference data from multiple grids, the power grid within which the recording was made can be identified since the overall trends of ENF fluctuation are the same within a grid. By examining detailed characteristics of the ENF and having multiple spatially located references, we may potentially narrow down the location within a grid [111], [113]. The ENF traces extracted from multiple streams may help forensically bind them, for example, to determine if the sound track and the visual track of a video were captured at the same time [109]. Inconsistencies observed in ENF, such as abrupt changes in extracted instantaneous frequencies and discontinuities in phase, signal potential content tampering [109], [114]. An example is shown in Fig. 10.

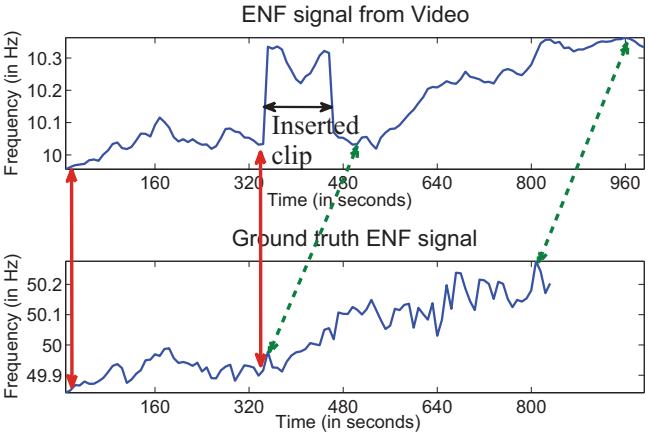


FIGURE 10. **ENF comparisons for video tampering detection.**

V. ANTI-FORENSICS AND COUNTERMEASURES

While existing forensic techniques are able to successfully detect multiple types of standard media manipulations, many forensic techniques do not account for the possibility that a forger with an advanced knowledge of signal processing techniques may attempt to disguise their forgery. As a result, researchers have begun studying *anti-forensic* operations capable of fooling forensic techniques.

This is important for a number of reasons. By developing anti-forensic operations, researchers can identify vulnerabilities in existing forensic algorithms. This will allow researchers to honestly assess how reliable their forensic results are. Furthermore, it may help researchers improve forensic algorithms. This will hopefully prevent scenarios in which forged multimedia files are declared authentic by forensic analysis. Additionally, anti-forensic operations may leave behind unique fingerprints in the same manner that traditional editing operations do. By studying anti-forensic operations, researchers can develop new techniques to detect the use of anti-forensics.

A. ANTI-FORENSIC TECHNIQUES

We start with providing a brief overview of existing anti-forensic research and the forensic techniques which they are designed to fool.

1) PRNU Forgery

As we discussed in Sections II and III, camera specific PRNU fingerprints that arise due to spatially distributed imperfections on a digital camera's imaging sensor can be used to identify which camera captured a digital image and verify an image's authenticity [84], [103]. An intelligent forger, however, can anti-forensically manipulate PRNU fingerprints, thus allowing a forger to disguise image forgeries and falsify a digital image's origin [115]. This is done by first removing the authentic PRNU left in an image using a technique called flatfielding. If the forger wishes to make an image appear

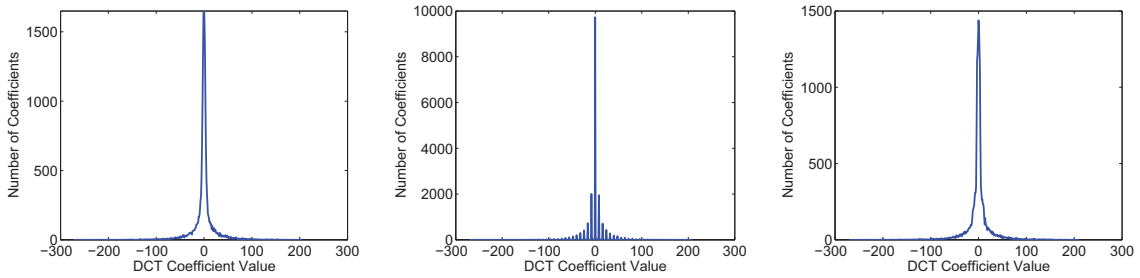


FIGURE 11. Histogram of coefficient values from the (2,2) DCT subband taken from an uncompressed image (left), the same image after JPEG compression (center), and an anti-forensically modified copy of the JPEG compressed image (right).

as if it were captured by another camera, the PNRU of the target camera can be estimated from a series images taken by the target camera. After this is done, the inauthentic PRNU is multiplicatively inserted into an image using inverse flat-fielding. Alternatively, a forger can modify the image, then re-insert an estimate of the authentic PRNU. This will prevent irregularities in the image's PRNU from revealing evidence of manipulation.

2) Hiding Traces of Image Resampling

Image resampling fingerprints can be hidden using a set of anti-forensic techniques [116]. This is especially important to forgers creating cut-and-paste image forgeries, because an object cut from one image must often be resized or rotated before it is pasted into another image. Since resampling fingerprints are caused by linear dependencies between pixels introduced by resampling, one simple approach to hiding these fingerprints is to apply a nonlinear filter such as a median filter. While nonlinear filtering can successfully hide resampling, it often introduces perceptually detectable distortion in the form of blurring. This problem is mitigated by a more sophisticated anti-forensic approach that uses a combination of nonlinear filtering and geometric distortion to remove resampling fingerprints. The strength of the geometric distortion is modulated by the local edge strength to ensure that it does not significantly degrade the quality of the anti-forensically modified image.

3) Falsifying an Image's Compression History

If the fingerprints of a particular processing operation can be accurately modeled, then this model can be used by a forger to design an anti-forensic technique to optimally remove these fingerprints. This has been done in the case of image compression fingerprints [117]. Recall from Section II-D that when an image is compressed using a transform coder such as JPEG, the quantization of its transform coefficients will leave behind fingerprints in the transform coefficients' distribution. By using a parametric model to describe the distribution of an uncompressed image's transform coefficients, the distribution of these transform coefficients after compression can be obtained by integrating this model over each quantization interval.

To remove the quantization fingerprints in transform coefficients, the model parameters of an uncompressed version of an image's transform coefficient distribution are first directly estimated from a compressed version of the image's transform coefficients. These parameters are used to design an anti-forensic dither, which is added to the quantized transform coefficients of the compressed image. The anti-forensic dither's distribution is chosen so that it matches the distribution of the uncompressed image's transform coefficients over each individual quantization interval. By doing this, it can be theoretically proven that an anti-forensically modified image's transform coefficient distributions will match those distributions before the image was compressed. This technique has been adapted for both standard JPEG compression as well as DWT based methods such as JPEG 2000 and SPIHT [117]–[119]. An example of the effectiveness of this anti-forensic technique can be seen in Fig 11, which shows DCT histograms from an image before compression, after JPEG compression, and after the JPEG compressed image has been anti-forensically modified.

Additionally, JPEG blocking artifacts can be erased by using median filtering followed by the addition of low power Gaussian noise to an image. The size of the median filter window and the variance of the additive noise can be adapted to the quality factor used during compression. This is necessary because existing deblocking techniques focus on removing visual traces of blocking fingerprints but do not completely remove all statistical blocking fingerprints [117], [120].

By using these anti-forensic techniques to remove both blocking fingerprints and DCT coefficient quantization fingerprints from JPEGs, a forger can disguise double JPEG compression, falsify an image's origin, and hide evidence of cut-and-paste forgeries [117], [120]. Double compression fingerprints can be prevented by anti-forensically removing DCT coefficient quantization fingerprints from an image before it is compressed a second time. If this is done to both images used when creating a cut-and-paste forgery, no localized mismatches in the forged image's compression history will arise. Blocking fingerprints can also be anti-forensically removed from each image to prevent the occurrence of misaligned blocking fingerprints in the cut-and-paste forgery. This will prevent forensic cut-and-paste detection techniques

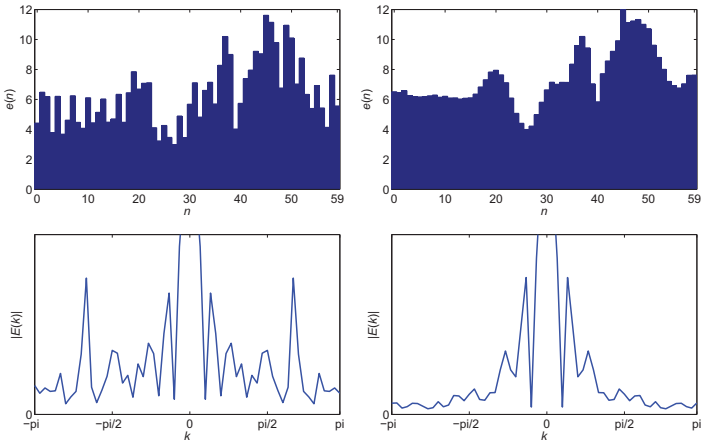


FIGURE 12. P-frame prediction error sequences (top row) and the magnitudes of their respective DFTs (bottom row) obtained from the 'Foreman' video (after the first six frames were deleted followed by recompression without anti-forensic modification (left column) and with the use of the anti-forensic technique in [68] (right column).

that use double compression or blocking fingerprints from identifying the forgery. Furthermore, an image's origin can be falsified by first erasing all previous JPEG compression fingerprints, then recompressing the image using the quantization table associated with a different digital camera. Additionally, the tradeoff between the compression rate, distortion introduced by anti-forensics, and the likelihood that a forgery will be concealed by anti-forensics has been recently charachterized [121].

4) Undetectable Video Frame Deletion

Similarly, fingerprints from frame deletion that are left in a digital video's sequence of P-frame prediction errors can be anti-forensically concealed by making use of the model of these fingerprints discussed in Section II-D [68], [122]. To accomplish this, the model of the video's P-frame prediction error sequence is used to identify spikes in the prediction error that correspond to frame deletion fingerprints. A feasible target sequence of P-frame prediction errors that does not contain frame deletion fingerprints is constructed by interpolating the values between these spikes using a cubic spline. Next, the prediction error in each P-frame is increased to its corresponding value in the target sequence by altering the motion vectors in each P-frame through an iterative process. Once an acceptable set of anti-forensically modified motion vectors is obtained, the prediction error for each P-frame is recalculated.

The effects of this anti-forensic algorithm can be seen in Fig. 12 which shows the P-frame prediction error sequence of a video before and after anti-forensic modification. It is important to note that altering the video in this manner results in no decrease in the visual quality of the anti-forensically modified video and ensures that the anti-forensically modified video can still be decoded by a standard video decoder.

5) Hiding Evidence of Contrast Enhancement

Contrast enhancement fingerprints can also be anti-forensically removed from a digital image. Recall that these fingerprints correspond to impulsive peaks and gaps in an image's pixel value histogram. These fingerprints can be removed by selectively adding Gaussian noise to pixels whose values lie in a certain neighborhood of pixel values corresponding to a peak or gap in the pixel value histogram. This will smooth out the peaks and gaps in the pixel value histogram that correspond to contrast enhancement fingerprints.

B. DETECTING ANTI-FORENSICS

In response to the development of digital anti-forensics, a small number of techniques designed to detect the use of anti-forensics have been recently developed.

1) PRNU Forgery Detection

If the PRNU of an image is forged to match the PRNU of a different camera, the PRNU of the target camera must be estimated from a set of images that the forger has access to. For example, these images may come from an online photo sharing website. As a result, the PRNU extracted from a forged image will be more highly correlated with the PRNU extracted from an image used to obtain the forged PRNU pattern than from a new image taken by the same camera. This fact can be exploited to create a technique capable of detecting PRNU forgery [123]. A linear relationship can be established between an estimate of the correlation score between an image's PRNU and a camera's reference PRNU, and the actual correlation score. This linear model will not hold when examining the correlation score between a forged image and an image used to estimate the forged PRNU pattern. Therefore, if a forensic examiner examines the estimated and actual correlation scores for at least one image used to obtain the forged PRNU, the forgery can be exposed.

2) Detecting Single JPEG Compression Anti-Forensics

Two sets of techniques have recently been proposed to identify images which have been anti-forensically modified to remove evidence of a single application of JPEG compression. One set of techniques operates by establishing a model of the relationship between the variance of different DCT subbands [124]. This model will likely not hold for DCT coefficients in high frequency subbands of anti-forensically modified images because it is difficult for a forger to accurately estimate the uncompressed distributions of these coefficients from the quantized DCT coefficients. Additionally, calibration techniques developed to perform steganalysis can be used to analyze the variance of each subband of DCT coefficients for inconsistencies. Alternatively, the use of anti-forensics can be detected by compressing an image in question at a variety of different JPEG quality factors and measuring the total variation of each compressed image [125]. If an image has been anti-forensically modified, the total variation will sharply increase for images compressed at higher quality factors than the original image.

It should be noted that while these techniques are able to identify when anti-forensics has been used to hide single JPEG compression, they do not work well when anti-forensics has been used to hide double JPEG compression.

3) Detecting Video Frame Deletion Anti-Forensics

While video frame deletion fingerprints can be prevented using anti-forensics, the anti-forensic technique used to do this leaves behind fingerprints of its own [68]. Recall that frame deletion anti-forensics operates by modifying a video's motion vectors in order to selectively increase the prediction error in certain P-frames and hide frame deletion fingerprints. Since the true motion in the video will not change due to anti-forensics, the use of anti-forensics can be exposed by analyzing a digital video's motion vectors and comparing them to the true motion in the scene. If the motion vectors stored in the video significantly differ from the true motion in the scene, then the video has likely been anti-forensically modified.

4) ENF Related Anti-Forensics and Countermeasures

As discussed in Section IV, the electrical network frequency (ENF) signal is a time stamp that is being used by an emerging class of approaches for determining the creation time and location of digital audio and video recordings. However, in adversarial environments, anti-forensic operations may be used to manipulate ENF-based time stamps. As a result, it is crucial to understand the robustness of ENF analysis against anti-forensics. A recent work [126] explored a set of possible anti-forensic operations that can remove and alter the ENF signal while trying to preserve the host signal, and developed detection methods targeting these anti-forensic operations. Concealment techniques that can circumvent detection were also discussed, and their corresponding trade-offs were examined. Based on the understanding of individual anti-forensic operations and countermeasures, the work provided an evolutionary view on the dynamic interplay between forensic analysts and adversaries. As with other anti-forensic problems, the understandings obtained on the strategy space of forensic analysts and adversaries can be incorporated into a game theory framework for quantitative analysis on the adversarial dynamics.

C. ADVERSARIAL DYNAMICS

Prior to the recent development of anti-forensics, the actions of a forger and forensic investigator were largely considered independently. It was generally assumed that a forger's primary concern was to create a perceptually realistic forgery. As a result, the detection strategy employed by a forensic investigator could be developed without having to account for an active adversary. The introduction of anti-forensic techniques, however, has highlighted a new dynamic interplay between a forger and a forensic investigator.

Since anti-forensic techniques enable a forger to actively evade forensic detection, the actions taken by a forger greatly influence the optimal detection strategy for a forensic investigator. Similarly, the detection strategy adopted by a forensic investigator must be considered when determining the forger's optimal anti-forensic countermeasures. As a result, each party must anticipate the actions of their adversary when identifying their own optimal course of action. In order to understand this dynamic interplay, researchers have recently begun using game theory to characterize the interactions between a forger and a forensic investigator.

To determine the optimal actions of a forger and forensic investigator if both the forger's use of a manipulating operation m and anti-forensics countermeasures α_m to hide m can be detected, a game-theoretic framework has recently been developed [68]. In this scenario, a forger can reduce the strength k of their anti-forensic technique to decrease the probability that their use of anti-forensics will be detected. This, however, will result in an increase in the probability that their manipulation will be detected. Since a forensic investigator's discovery of either either file manipulation or the use of anti-forensics will expose a forgery, the forger must identify the anti-forensic strength that minimizes the investigator's overall probability of detecting their forgery.

On the other hand, a forensic investigator must balance a tradeoff of their own caused by a constraint on their total false alarm rate. Since the investigator's manipulation detection algorithm and anti-forensics detection algorithm will both contribute to the total false alarm rate, the investigator must determine how much each algorithm will contribute to the total false alarm rate. This is critical because the performance of each detector is dependent on the false alarm rate allocated to it. The false alarm rate allocation that maximizes the investigator's probability of identifying a forgery, however, is dependent upon the anti-forensic strength used by the forger.

To identify the actions that both parties are incentivized to take, the action set of the forger and investigator are characterized as the set of anti-forensic strengths and false

alarm rate allocations respectively. Next, the utility function that the forensic investigator wishes to maximize is defined as the probability that a forgery will be identified either by detecting manipulation or the use of anti-forensics. Similarly, the utility function for the forger is defined as the negative of the investigator's utility minus a term to account for the cost of introducing perceptual distortion into their forgery. Using these utility functions, a set of Nash equilibrium strategies that neither party will have any incentive to deviate from can be identified.

This game-theoretic analysis leads to a technique for measuring the overall performance of the forensic investigator. By varying the investigator's total false alarm constraint 0% and 100% and identifying the Nash equilibrium strategies at every point, the probability that the investigator detects a forgery under equilibrium can be determined at all possible false alarm rates. These can be combined together to create a Nash equilibrium receiver operating characteristic (NE ROC) curve.

This framework has been used to analyze the interaction between a forger and investigator in the context of video frame deletion forensics. The NE ROC characterizing this scenario was determined using the results of this analysis and it was discovered that if a forensic investigator is constrained to operate with a false alarm rate of less than 10%, it is likely that video frame deletion will not be detected. Conversely, if the investigator is allowed to operate with a false alarm rate of 15% or above, it is highly likely that any attempt to delete frames from a video will be detected.

Additionally, game theory has been used to investigate adversarial dynamics in the context of identifying a multimedia file's source [127]. In this scenario, a forensic investigator must identify the optimal strategy for determining the source of a sequence of multimedia files. Similarly, a forger will attempt to use the optimal anti-forensic strategy to falsify the source of these multimedia files. The asymptotic Nash Equilibrium can be identified for an infinitely long sequence of media files, and this information can be used to approximate the investigator and forger's optimal forensic and anti-forensic strategies respectively.

VI. EMBEDDED FINGERPRINTING AND FORENSIC WATERMARKING

As we have seen so far in this article, to perform forensics on multimedia, one must start with discovering some traces of evidence. Analogous to human fingerprints in criminal forensics, there are two forms of "fingerprints" in digital domain. The previous sections have reviewed invisible traces of evidence left in content when the content goes through various devices and operations. These "intrinsic" fingerprints are shown to provide powerful forensic evidence regarding the history and provenance of digital content, as well as the inner workings of the associated devices. "Extrinsic" approaches complement "intrinsic" ones to provide proactive protections through embedded or attached data. Earlier work on extrinsic techniques in the late 1990s focused on embedding information to represent ownership rights or verify host media data's integrity [12], [13]. In this section, we discuss advances in "extrinsic" fingerprints in the most recent decade. These advances extend traditional robust watermarking to embed a unique ID or signal representing each user in his or her copy of a multimedia file [128]–[130], and use this later as a forensic tracer in the event of leak or misuse to determine who was the original recipient. Together, these forensic technologies complement each other and can provide useful evidence to help answer a number of forensic questions arising from law enforcement, intelligence operations, journalism, and intellectual property protection.

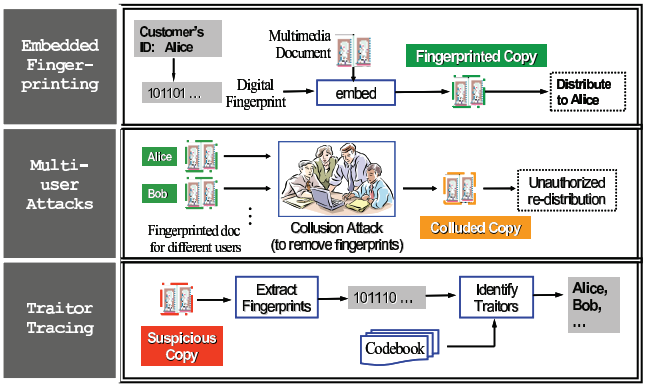


FIGURE 13. Using embedded fingerprinting for tracing users. (Source: [129])

A. BACKGROUND: ROBUST EMBEDDING AND COLLUSIONS

Fig. 13 shows an overview of the process involving embedded fingerprinting for tracing content distributions. As with other forensic problems, adversaries have incentives to apply a variety of attacks to circumvent fingerprint detection from a multimedia file. An adversary may act alone to apply filtering, compression, and other noise and distortions in hopes to remove or weaken fingerprints, or render them difficult to be detected with high confidence. *Collusion* is a powerful multi-user attack where a group of users combine their copies of the same content to generate a new version with the fingerprint attenuated or removed. In addition to resistance against attacks, three aspects of system efficiency need to be considered when designing an anti-collusion fingerprinting system, namely, the efficiency in constructing, detecting and distributing fingerprinted signals. Construction efficiency concerns the computational complexity involved during the generation of fingerprinted content; detection efficiency is related to the computational complexity of detection; distribution efficiency refers to the amount of bandwidth consumed during the transmission of all the fingerprinted signals.

1) Spread-Spectrum Based Robust Embedding

An embedded fingerprinting system for the tracing and tracking of multimedia data builds on top of robust data embedding methods that are capable of withstanding adversaries' attacks to remove the embedded fingerprint. In this aspect,

embedded fingerprinting is closely related to digital watermarking and the fingerprints are sometimes referred to as ''forensic watermarks''. In contrast to having a single marked copy available to adversaries in most watermarking applications, the presence of multiple distinctly marked copies in most
fingerprinting applications prompts the additional challenge of creating collusion resilient designs to combat collusion attacks and identify colluders. Many techniques have been proposed for embedding information in multimedia signals [12]. Here we briefly review the spread spectrum additive embedding technique and its role in robustly embedding fingerprint signals into multimedia.

Spread spectrum embedding borrows ideas from spread spectrum modulation [137]. The basic embedding process consists of the following steps. The first step is to identify and compute proper features that will carry watermark signals. Depending on the design requirements and the type of media data, the features can be signal samples, transform coefficients (such as DCT and wavelet coefficients), or from other functions suitable for the specific type of media content (such as local shape parameters for curves [138]). Next, we generate a watermark signal and apply perceptual models to tune its strength to ensure imperceptibility. A spread spectrum watermark typically resembles weak noise and covers a broad spectrum as well as a large region of the media data. Finally, we add the watermark to the feature signal, replace the original feature signal with the watermarked version, then convert it back to the signal domain to obtain a watermarked signal. The detection process for spread spectrum watermarks begins with extracting features from a media signal in question. Then the similarity between the features and a watermark is examined to determine the existence or absence of the specific watermark in the media signal. A correlation similarity measure is commonly used, often in conjunction with preprocessing (such as whitening) and normalization to achieve reliable detection [12].

Spread spectrum embedding has been shown to be very robust against a number of signal processing operations and attacks [131], [132], including strong lossy compression, format conversions, filtering, and more. By appropriately choosing features and applying alignment procedures, spread spectrum watermarks can survive moderate geometric distortions, such as rotation, scaling, shifting, and cropping [133], [134]. Information theoretic studies also suggest [135], [136] that it is nearly capacity optimal when the original host signal is available in detection, which is a situation often considered in fingerprinting applications.

2) Orthogonal vs. Code Modulation

A simple way to realize fingerprinting via spread spectrum embedding is to use mutually orthogonal watermarks as fingerprints to identify each user [139], [140]. The orthogonality allows for distinguishing the fingerprints to the maximum extent. The simple encoding and embedding of orthogonal fingerprints makes them attractive to applications that involve a small group of users and tractable for analytic studies on their performances [141]–[143]. The orthogonality may be approximated by using pseudo-random number generators to produce independent watermark signals for different users.

A second option of using spread spectrum watermarking is to employ code modulation [144], [145]. Code modulation allows fingerprint designers to design more fingerprints for a given fingerprint dimensionality by constructing each user's fingerprint signal as a linear combination of orthogonal noise-like basis signals. For a large number of users, the detection complexity of coded fingerprinting can be much lower than that of the orthogonal construction that is proportional to the number of users. To embed a codeword, we first partition the host signal into L non-overlapped segments, which can be one block of image or a frame or a group of frames of video, with one segment corresponding to one symbol. The fingerprint signal for the j^{th} user, $\mathbf{w}_j$, is constructed using a linear combination of a total of v orthogonal basis signals $\{\mathbf{u}_i\}$:

$$\mathbf{w}_j = \sum_{i=1}^{v} b_{ij}\mathbf{u}_i. \tag{3}$$

Here the coefficients $\{b_{ij}\}$, representing fingerprint codes, are constructed by first designing codevectors with values {0, 1}, and then mapping them to $\{\pm 1\}$.

It is possible to extend the binary coded fingerprinting to M-ary by employing M orthogonal basis signals for each M-ary code symbol [145], [146]. For each segment, we generate M mutually orthogonal spread spectrum sequences $\{\mathbf{u}_1, ..., \mathbf{u}_M\}$ with equal energy to represent the M possible symbol values in the alphabet. Each user's fingerprint sequence is constructed by concatenating the spreading sequences corresponding to the symbols in his/her codeword. The fingerprint sequence is then perceptually shaped and added to the host signal through spread spectrum embedding to form the final fingerprinted signal.

3) Collusion Attacks

As mentioned earlier, the collusion attack is a powerful collaboration attack by multiple users. During a collusion attack, a group of colluders holding differently fingerprinted versions of the same content examine their different copies and attempt to create a new signal that would be difficult to be linked to any of the colluders. There are several types of collusion attacks [129], [147]. *Linear collusion attack* is simply to align the fingerprinted copies and take a linear combination of their corresponding elements. Another collusion attack, referred to as the *copy-and-paste attack* or *interleaving attack*, involves users taking parts of their media signals and assemble these parts together into a new version. Other attacks may employ non-linear operations, for example, following order statistics such as taking the maximum or median of the values of corresponding components of individual copies. Considering a simple fairness among colluders each of whom would not want to assume higher risk of being caught than others, each colluder would contribute a similar amount of share, for

example, by averaging their signals, leading to the *averaging collusion attack*.

Research has shown that for orthogonal fingerprinting based on the spread-spectrum embedding, interleaving collusion and many variants of order statistics based nonlinear collusion have a similar effect on the detection of fingerprints to the collusion by averaging and possibly followed by additive noise [148], [149]. So if the overall distortion level introduced by collusion attacks is the same, similar detection performance can be observed although the specific form of collusion may be different. If the fingerprint is constructed in modules such as through coding, the simple collusion model of averaging plus additive noise can also approximate many collusion attacks for systems where all code components are spread over the hosting signal in such a way that adversaries can only distort them as a whole and cannot alter them individually [150]. On the other hand, for many code based systems whereby various code segments of the fingerprints are embedded in different parts of the hosting signal, different collusion attacks may have different levels of effectiveness [145]. We shall discuss more on this in the next subsection.

It is worth mentioning that *intra-content collusion* may be mounted against fingerprints by a single user by replacing each segment of the content signal with another, seemingly similar segment from different spatial or temporal regions of the content [151]. This is particularly of concern in the protection of video signals, whereby adjacent frames within a scene appear very similar to each other. A basic principle to resist these attacks is to embed fingerprint sequences based on the content of the video [152], so that similar fingerprint sequences are embedded in frames or objects with similar content and different fingerprint sequences are embedded in frames with different content. A visual hash of each frame may be employed to adjust the fingerprint sequences [153].

It is relatively rare to see the notion of ''anti-forensics'' in the literature on embedded fingerprinting (unlike the growing interests in anti-forensics in the context of non-intrusive/intrinsic fingerprinting discussed in the earlier section). This can be in part due to the fact that the considerations of adversaries and attacks are often closely interwoven into the problem formulation, such as collusion models. In the next section, we shall also see the investigation into analyzing adversaries' behaviors in the context of embedded fingerprinting.

B. COLLUSION RESISTANT CODED FINGERPRINTING

As reviewed above, a typical framework for code based multimedia fingerprinting includes a code layer and a spread spectrum based embedding layer. A number of code designs have been discussed in the literature and they achieve a varying degree of collusion resilience and efficiency. Among then, randomized codes, combinatorial design, and error correcting codes are major representatives.

1) Marking Assumptions and Random Codes

An early work on designing collusion-resistant binary fingerprint codes was presented by Boneh and Shaw in 1995 [154], which considered the problem of fingerprinting generic data under the *Marking Assumption* that governs what adversaries can and cannot do. In this work, a fingerprint consists of a collection of marks, each of which is modeled as a position in a digital object and can have a finite number of different states. A mark is considered detectable when a coalition of users does not have the same mark values in that position. The Marking Assumption states that undetectable marks cannot be arbitrarily changed without rendering the object useless; however, it is considered possible for the colluding set to change a detectable mark to any state. Under this collusion framework, Boneh and Shaw used hierarchical design and randomization techniques to construct *c-secure codes* that are able to capture one colluder out of a coalition of up to c colluders with high probability. Interested readers may refer to [129] for an tutorial example on the construction of *c-secure codes*.

The construction strategies of Boneh-Shaw codes offers insight into fingerprinting both bitstreams and other data for which each bit or unit is marked with a fingerprint component in a non-overlapping manner. Improvement was introduced in [155] to merge the low-level code with the direct sequence spread spectrum embedding for multimedia and to extend the Marking Assumption to account for random jamming.

Tardos' Code is an alternative design to the Boneh-Shaw code and has received growing attention in recent years [156], [157]. Tardos' code is a class of random binary codes, whereby the i^{th} bit follows a Bernoulli distribution with parameter P_i that is a random variable itself symmetrically distributed about $1/2$. The key of Tardos' code is to optimally design the distribution for P_i and the decision functions to process a colluded version of the code depending on the matches and unmatches with a given user's bit values.

Multimedia data have some unique characteristics different from generic data, and a few fundamental aspects of the Marking Assumption may not always hold when fingerprinting multimedia data. For example, different ''marks'' or fingerprint symbols can be embedded in overlapped regions of an image through spread spectrum techniques, and such ''overlapped spreading'' can make it impossible for attackers to know and manipulate individual marks at will. As a result, other collusion models, such as linear collusion by averaging, should be considered for multimedia fingerprints. This has a critical impact on designing fingerprint codes.

2) Anti-Collusion Codes for Multimedia via Joint Combinatorial Designs and Embedding

Research in [144] explored the joint consideration of the encoding, embedding, and detection of fingerprints for multimedia, and developed a class of combinatorial theory based codes, known as *anti-collusion codes (ACC)*, in conjunction with spread spectrum code modulation for

User 1:	-1	-1	-1	-1	1	1	1	1	1	1	1	1	1	1	1	1
User 4:	-1	1	1	1	1	1	1	1	1	1	-1	-1	-1	1	1	1
User 8:	1	-1	1	1	1	1	-1	1	-1	1	1	1	1	1	-1	1
User(1,4) Average:	-1	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1
User(1,4,8) Average:	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	1	1	$\frac{1}{3}$	1	$\frac{1}{3}$	1	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	1	$\frac{1}{3}$	1
After thresholding:	0	0	0	0	1	1	0	1	0	1	0	0	0	1	0	1

FIGURE 14. 16-bit ACC codevectors for user 1, 4, and 8, and the fingerprinted 512 × 512 Lenna images for these three users, respectively. The code can capture up to 3 colluders. Shown here is an example of two-user collusion by averaging (user 1 and 4) and an example of three-user collusion by averaging. The two codes indicated by arrows in the table uniquely identify the participating colluders. (Source: [129])

embedding. An anti-collusion code is a family of codevectors for which the bits shared between codevectors uniquely identifies groups of up to K colluding users, as the composition of any subset of K or fewer codevectors is unique. A K-resilient AND anti-collusion code (AND-ACC) is such a code where the composition is an element-wise AND operation. It has been shown that binary-valued AND-ACC exist and can be constructed systematically using balanced incomplete block designs (BIBD) from combinatorial mathematics [144]. When two watermarks are averaged, the set of locations where the corresponding AND-ACC agree and have a value of 1 is unique with respect to the colluder set, and can therefore be used to identify colluders. Fig. 14 shows an example of a 16-bit ACC using BIBD construction, and the collusion effect of two- and three- user collusion, respectively. Another attempt at using the theory of combinatorics to design fingerprints was made by [158], where projective geometry was used to construct the codes.

3) Fingerprinting Based on Error Correcting Codes (ECC)

A group of coded fingerprinting extends Boneh and Shaw's framework and considers the construction of codes with traceability and affordable identification algorithms in terms of computational complexity [159]. Fingerprint codes that can be systematically constructed using well-established error correcting codes (ECC) are of particular interests, and a large minimum distance of the codes ensures that the fingerprint codewords for different users are well separated [160]. The two-level code construction from the aforementioned Boneh-Shaw's work was extended to incorporate ECC in [161], which uses the orthogonal fingerprinting in the low level and structured error correction code in the upper level to improve the detection efficiency over the traditional single-level orthogonal fingerprinting.

A design methodology involving ECC is to treat the symbols contributed from other colluders as errors, and makes the minimum distance between codewords large enough to tolerate the errors. The minimum distance requirement ensures that the best match with a colluded codeword (referred to as the descendant) comes from one of the true colluders. The traceability code for resisting c colluders, or c-TA code in short, is such an example [162]. Under the attack model by interleaving collusion, a c-TA code can be constructed using an ECC if its minimum distance D satisfies [162]

$$D > (1 - \frac{1}{c^2})L \tag{4}$$

where L is the code length and c is the colluder number. As the distortions and attacks mounted by adversaries on the fingerprinted multimedia data can lead to errors in detecting fingerprint code symbols, these errors and erasures should be accounted for and the above code parameter requirements can be extended accordingly [146]. To tolerate more colluders calls for an ECC with a larger minimum distance. As Reed-Solomon codes have the minimum distance that achieves the Singleton bound [163], it has been a popular choice in the literature [146], [160], [162].

As we can see, because of a relatively small alphabet size q compared to the number of users N_u and also owing to one symbol being put in one non-overlapping media segment, ECC fingerprinting has the advantage of being able to generate and distribute fingerprinted media in an efficient way [153]. For example, for each frame of a video, a total of q copies carrying q different symbol values can be generated beforehand. A fingerprinted copy for any user can then be quickly obtained by assembling appropriate copies of the frames together according to the fingerprint codeword assigned to the user. The small alphabet size also keeps the computational complexity of fingerprint detection lower than the orthogonal fingerprinting approach [145].

The collusion resistance performance of a straightforward embedding of ECC-based fingerprint code is, however, rather limited. Consider an ECC based fingerprinting system

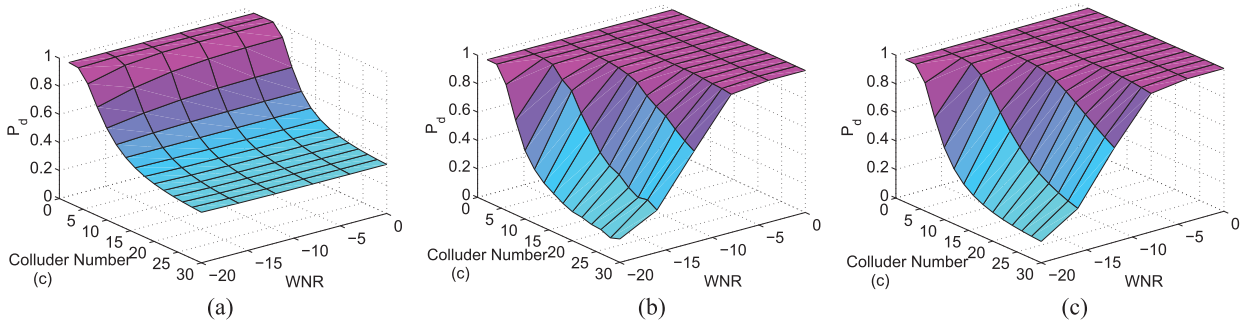


FIGURE 15. Fingerprint detection performance of simple ECC based fingerprinting as a function of colluder number and watermark-to-noise ratio (WNR) under (a) interleaving collusion; (b) averaging collusion; and of orthogonal fingerprinting under (c) interleaving collusion. (Source: [145])

employing a L-tuple code with minimum distance D over q-ary alphabet to represent N_u users. Under the (symbol wise) interleaving collusion, colluders can exploit the fingerprint pattern and contribute segment by segment with each segment carrying one symbol. In contrast, averaging collusion does not rely on the fingerprint pattern and simply takes the average value of each signal component, thus allowing each of the L symbols from every user to leave some traces on the final colluded signal. These two collusion attacks have different effects on colluder detection, and can be analyzed based on signal detection theory [145]. As illustrated in Fig. 15, the studies have revealed a significantly lower collusion resistance under interleaving collusion than that under averaging collusion and that of orthogonal fingerprinting under either type of collusion.

Such a drastic difference in the collusion resistance against averaging and interleaving collusions of ECC based fingerprinting inspired an improved fingerprinting method incorporating randomization [145], for which the interleaving collusion would have a similar effect to averaging collusion. This was motivated by the observation that the segment-wise interleaving collusion is equivalent to the symbol-wise interleaving collusion on the code level, since each colluded segment comes from just one user; thus the collusion resilience primarily relies on what is provided by the code layer and the role from the embedding layer is minimal. The limited alphabet size makes the chance for the colluders' resulting fingerprint via interleaving becoming close to an innocent user's fingerprint so high that it would require a large minimum distance in the code design if it is to be handled on the code level alone. This means that either codes representing a given number of users can resist only a small number of colluders, or codes can represent only a small total number of users. On the other hand, for averaging collusion, the embedding layer contributes to defending against the collusion through the use of a correlation detector. A solution to this problem builds upon the existing code construction and performs additional steps that are collectively refer to as *Permuted Subsegment Embedding* [145]. With subsegment partitioning and permutation, each colluded segment after interleaving collusion most likely contains subsegments from multiple users, and the added randomness makes it very difficult for colluders to choose the resulting combinations at will. To correlation-based detectors, this would have a similar effect to what averaging collusion brings for which colluders' effectiveness is low, thus the overall collusion resistance of ECC based fingerprinting can be significantly improved. This further demonstrates the benefit from jointly considering fingerprint encoding, embedding, and detection.

4) Fingerprint Multicast in Secure Streaming

As discussed earlier, there are two main issues with multimedia fingerprinting systems. First, collusion is a cost effective attack, where several users (colluders) combine several copies of the same content but embedded with different fingerprints, and they aim to remove or attenuate the original fingerprints. Second, the uniqueness of each copy poses new challenges to the distribution of fingerprinted copies over networks, especially for video streaming applications where a huge volume of data have to be transmitted to a large number of users. Video streaming service providers aim to reduce the communication cost in transmitting each copy and, therefore, to accommodate as many users as possible, without revealing the secrecy of the video content and that of the embedded fingerprints. The work in [164] addresses the second issue concerning secure and bandwidth efficient distribution of fingerprinted copies.

A simple solution of unicasting each fingerprinted copy is inefficient since the bandwidth requirement grows linearly as the number of users increases, and the difference between different copies is small. Multicast provides a bandwidth advantage for content and network providers when distributing the same data to multiple users. It reduces the overall communication cost by duplicating packages only when routing paths to multiple receivers diverge. However, traditional multicast technology is designed to transmit the same data to multiple users, and it cannot be directly applied to fingerprinting applications where different users receive slightly different copies. This calls for new distribution schemes for multimedia fingerprinting, in particular, for networked video applications.

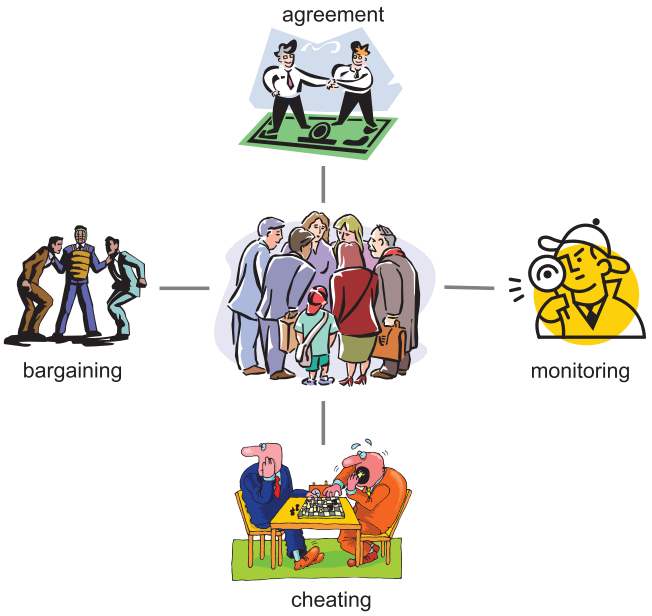


FIGURE 16. User dynamics in social networks.

In spread spectrum embedding, not all coefficients are embeddable due to the perceptual constraints on the embedded fingerprints, so the values of a nonembeddable coefficient in all copies are identical. To reduce the communication cost in distributing these nonembeddable coefficients, a general fingerprint multicast scheme was developed in [164] that multicasts the nonembeddable coefficients to all users and unicasts the uniquely fingerprinted coefficients to each user. This scheme can be used with most spread spectrum embedding-based fingerprinting systems. To further reduce the bandwidth in distributing these fingerprinted coefficients, a joint fingerprint design and distribution scheme was also developed to multicast these shared fingerprinted coefficients to the users in that subgroup. Such a joint fingerprint design and distribution scheme utilizes the special structure of the fingerprint design for higher bandwidth efficiency.

VII. BEHAVIOR/HUMAN/SOCIAL FACTORS AND DYNAMICS IN FORENSICS

The advances of broadband networking allows efficient, scalable and robust distribution and sharing of multimedia content over large and heterogeneous networks. The content owners, users, as well as attackers basically form a social group or network that interacts with each other, as illustrated in Fig. 16. A crucial issue is to understand how users interact with and respond to each other, and analyze the impact of human factors on information forensics. Such an understanding provides fundamental guidelines to better the design of multimedia systems and networking, and to offer more secure and personalized services. Therefore, protecting digital contents is no longer a traditional security issue with a single adversary. The global nature of the Internet has enabled a group of attackers (colluders) to work together and collectively mount attacks to remove their fingerprints or traces of evidences. These attacks, such as the multiuser collusion discussed in the previous section, pose serious threats to intellectual property rights. Analysis of the strategies, capabilities and limitations of attackers is an indispensable and crucial part of multimedia forensics research.

A. BEHAVIOR MODELING AND FORENSICS

In multimedia fingerprinting, different players have different goals and they influence each otherŠs decisions and performance. During collusion, attackers form a unique social network: they share the reward from the illegal usage of multimedia as well as the risk of being captured by the digital rights enforcer. An agreement must be reached regarding how to distribute the risk and the reward before collusion relationship can be established. However, each colluder prefers the agreement that favors his or her payoff the most, and different colluders have different preferences. To address such a conflict, a critical issue is to decide how to fairly distribute the risk and the reward. In addition, even if all colluders agree on how to distribute the risk and reward, some colluders might be selfish and wish to break away from their fair-collusion agreement. They might cheat to their fellow attackers during the negotiation process in order to minimize their own risk and maximize their own payoff. On the other hand, to protect their own interest, other colluders may want to identify selfish colluders and exclude them from collaboration. It is important to understand how colluders negotiate with each other to achieve fairness of the attack and study the cheating and cheat-proof strategies that colluders may adopt to maximize their own payoff and protect their own interest.

To maximize their own payoff, users should observe and learn how others play the game and adjust their own strategies accordingly. For example, to maximize the traitor-tracing capability, the digital rights enforcer should explore and utilize as much knowledge about collusion as possible when designing the fingerprints and identifying the colluders. Here, analyzing the colluder dynamics, especially the investigation on how attackers achieve fairness of collusion, provides the digital rights enforcer with important insights on how to probe and use such side information about collusion. Therefore, another important issue in behavior modeling is to understand the techniques that users can use to probe information about how others play the game, study how they adjust their strategies accordingly to maximize their own payoff, and analyze the impact of side information on multimedia social networks.

B. FAIRNESS DYNAMICS

Taking the view point of users who take part in collusion, we see that by contributing their own resources and cooperating with each other, colluders are able to access extra resources from their peers and thus receive rewards. Each user aims to maximize his or her own payoff and different users have different (and often conflicting) objectives. To address this conflict, an important issue is to investigate users' strategies to achieve a notion of fairness [165].

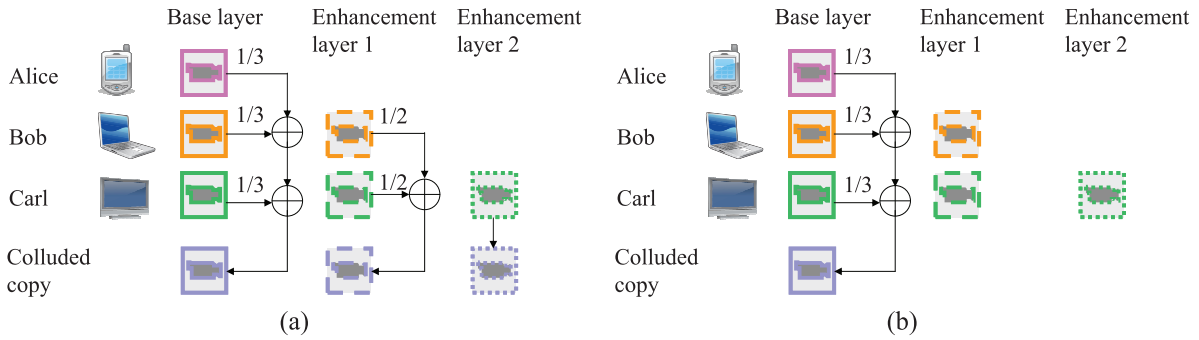


FIGURE 17. Two solutions of collusion in scalable multimedia fingerprinting.

Colluders receive rewards from the illegal usage of multimedia content, for example, the profit from the unauthorized redistribution of copyrighted materials. They also take the risk of being captured by the digital rights enforcer, which can be considered as the colluders' "cost" by participating in collusion. The notion of equal-risk absolute fairness is widely adopted in the literature, where all colluders agree to share the same risk and have equal probabilities of being detected. When all colluders receive fingerprinted copies of the same resolution, a simple average of all copies with equal weights reduces the energy of each contributing fingerprint by the same ratio, and it as well ensures the equal risk of all attackers.

When colluders receive fingerprinted copies of different resolutions, it is much more complicated to guarantee the equal risk of all colluders, especially when colluders wish to generate a colluded copy of higher resolution. For the example with three colluders, Alice, Bob and Carl, who receive fingerprinted copies of different resolutions, a possible solution of collusion was shown in Fig. 17(a), where the colluded copy includes all three layers. Here, the colluders average the three base-layer copies that they have with equal weights 1/3; for the enhancement layer 1, they average the two copies from Bob and Carl with equal weights 1/2; and the colluded copy's enhancement layer 2 equals to that in Carl's copy. Therefore, in the colluded copy, the three fingerprints corresponding to the three attackers have the same energy in the base layer, while the enhancement layers contain only Bob and Carl's fingerprints and not the fingerprint identifying Alice. It is obvious that among the three, Carl has the largest probability of being caught and Alice takes the smallest risk. Consequently, the collusion in Fig. 17 (a) does not achieve equal-risk fairness.

Fig. 17(b) shows another possible solution, where the colluded copy contains the base layer only. Here, the colluders average the three copies of the base layer with equal weights 1/3. In this example, the fingerprints corresponding to the three attackers have the same energy in the colluded copy and, therefore, the three attackers have the same probability of being detected. Although the collusion in Fig. 17(b) ensures equal-risk fairness, the attacked copy has a low resolution.

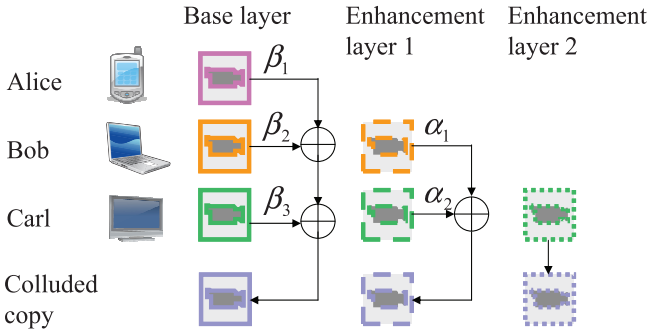


FIGURE 18. The intra-group and the inter-group collusion attacks.

Now the question is: when there is a difference in the resolution of fingerprinted copies due to network and device heterogeneity, how can colluders establish fair multiuser collusion that guarantees the collective equal risk among all attackers while still generating an attacked copy of high resolution? A possible solution is shown in Fig. 18. In the base layer of the colluded copy, the three copies are assigned different weights β_1, β_2 and β_3, respectively. Similarly, the enhancement layer 1 in the colluded copy is the average of Bob and Carl's copies with weights α_1 and α_2, respectively. The colluders simply copy the enhancement layer 2 in Carl's copy to the colluded copy. To achieve fairness of collusion, Alice, Bob and Carl select the collusion parameters $\{\alpha_k, \beta_l\}$ such that they have the same probability of being detected.

C. GAME-THEORETIC MODELING OF COLLUDER DYNAMICS

Equal-risk absolute fairness only considers each colluder's risk and ensures that all colluders have the same probability of being detected. During collusion, colluders not only negotiate how to distribute the risk but also bargain how to share the rewards from the illegal usage of multimedia. In addition, rather than absolute fairness, colluders may prefer other ways to distribute the risk and the reward. For example, some colluders may want to benefit more from collusion by taking a higher risk of being detected. In [166], [167], this complex dynamics was modeled as a bargaining problem where

colluders negotiate with each other to resolve the conflict, and game theory [168] was used to analyze this negotiation process.

In this game-theoretic framework, colluders first define the utility (payoff) function π, which is a function of a colluder's risk as well as the reward that he or she receives from collusion. A natural definition of the utility function is the expected payoff that a colluder receives by participating in collusion. For colluder $\mathbf{u}^{(i)}$, his or her utility can be given by

$$\pi^{(i)} = -P_s^{(i)} L^{(i)} + (1 - P_s^{(i)}) R w^{(i)} \tag{5}$$

where $P_s^{(i)}$ is his or her probability of being detected, $L^{(i)}$ is colluder $\mathbf{u}^{(i)}$'s loss if he or she is captured by the fingerprint detected, and $Rw^{(i)}$ is the reward that $\mathbf{u}^{(i)}$ receives if he or she successfully escapes being detected. Each colluder tries to maximize his or her own utility function during the negotiation process.

Without loss of generality, we use a two-layer multimedia fingerprinting example to demonstrate how colluders bargain during collusion. We assume that there are a total of 250 colluders, among which 80 attackers receive the low-resolution copies and the rest have the high-resolution version. For simplicity, we consider a scenario where colluders who receive fingerprinted copies of the same resolution agree to share the same risk and have equal utilities. Therefore, colluders who receive the low-resolution copies act as a single player in the game and they have the same utility π^b; while colluders who have the high-resolution copies act as a single player during the bargaining process and they have the same utility π^{be}.

The second step in the bargaining process is to find the feasible set $\mathbf{S} = \{(\pi^b, \pi^{be}) \in \mathbb{R}^2\}$ of the game, where for every $(\pi^b, \pi^{be}) \in \mathbf{S}$, it is possible for colluders to act together and obtain the utilities π^b and π^{be}, respectively. For the above mentioned colluder game, Fig. 19 shows the feasible set, which is the curve **AB** plus the line **BC**. Note that if colluders select a solution that corresponds to one point on the line **BC**, then they can always find another solution that gives the same π^{be} but a larger π^b. Therefore, in a bargaining situation like this, colluders would always like to settle at a Pareto-Optimal point, where no one can further increase his or her utility without decreasing others'. In Fig. 19, the Pareto-Optimal set includes solutions that correspond to the points on the curve **AB**.

Depending on their definition of fairness and their objectives of collusion, colluders select different collusion strategies. For example, with *equal-payoff absolute fairness*, colluders select the point where $\pi^b = \pi^{be}$ and let all attackers have the same utility. Colluders can also select the collusion parameters to *maximize the minimum utility* that a colluder can receive by participating in collusion, that is,

$$\pi^* = \max_{\beta} \min\{\pi^b, \pi^{be}\} \tag{6}$$

where β is the collusion parameter in Fig. 18. This solution guarantees that by participating in collusion, a colluder can receive at least π^* utilities. The *maxsum solution* maximizes

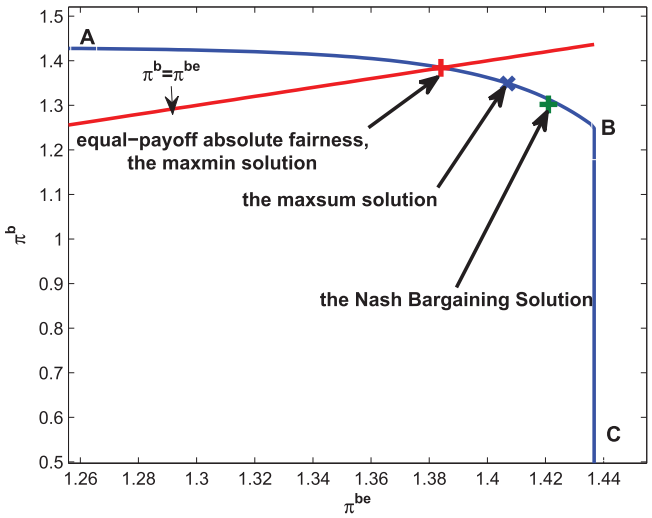


FIGURE 19. An example of the feasible set and different solutions of the colluder game. The X axis is the utility of colluders who receive the high-resolution copies, andthe Y axis is the utility of colluders who receive the low-resolution copies.

the sum of all attackers' utilities if they cooperate with each other during collusion. Another popular solution in game theory is the famous *Nash Bargaining Solution* (NBS), which aims to achieve *proportional fairness*. It divides the additional utility between the two players in a ratio that is equal to the rate at which this utility can be transferred [168]. Mathematically, the Nash Bargaining Solution maximizes

$$g(\pi^b, \pi^{be}) = \left(\pi^b - \pi^{b*}\right)\left(\pi^{be} - \pi^{be*}\right),$$
$$\text{where} \quad \pi^{b*} = \min_{\beta}\{\pi^b\} \quad \text{and} \quad \pi^{be*} = \min_{\beta}\{\pi^{be}\}. \tag{7}$$

Different collusion strategies correspond to different points in the Pareto-Optimal set. In the example shown in Fig. 19, the equal-payoff absolute fairness and the maxmin strategies give the same result, while the maxsum and the Nash Bargaining solutions favor colluders who receive the high-resolution fingerprinted copies more.

D. RELATED WORKS

Among the collusion attackers, it is necessary to give each other correct information about their own resources to achieve fairness. However, the assumption of fair play may not always hold. Although they might agree so, some users might be selfish and wish to maximize their own payoff. To achieve this goal, they might break away from their agreement and cheat to other users during the bargaining process. To improve the overall system performance, it is important to study the cheating and cheat-proof dynamics among users and investigate the selfish colluders' cheating strategies, and design cheat-proof mechanisms. In [169], multiuser collusion is used as an example to understand the colluders' cheating and cheat-proof strategies and study the "traitor-within-traitor" problem. It formulated the dynamics among attackers and understand their behavior to minimize their own risk and protect their own interests. It also explored some possible

strategies that a selfish colluder can use to minimize his or her probability of being caught. It was shown that processing his or her fingerprinted copy before multiuser collusion helps a selfish colluder further lower his or her risk, especially when the colluded copy has high resolution and good quality. The optimal precollusion processing strategies were considered for selfish colluders to minimize their risk under the quality constraints.

Another work [170] investigated human behavior dynamics in multimedia social networks with side information. Side information is information other than the colluded multimedia content that can help increase the probability of detection. It showed that the statistical means of the detection statistics can help the fingerprint detector significantly improve the collusion resistance. It then investigated how to probe the side information and model the dynamics between the fingerprint detector and the colluders as a two-stage extensive game with perfect information. By modeling the colluder-detector behavior dynamics as a two-stage game, it found the equilibrium of the colluder-detector game using backward induction. Additionally, it showed that the min-max solution is a Nash equilibrium which gives no incentive for anyone in the multimedia fingerprint social network to deviate. This work demonstrated that side information can significantly help improve the system performance so that it is almost the same as the optimal correlation-based detector. This result opens up a new paradigm in fingerprinting system research: given any fingerprint code, leveraging side information can improve the collusion resistance. It also provided the solutions to reach optimal collusion strategy and the corresponding detection, thus lead to a better protection of the multimedia content.

As we understand now that each user wants to maximize his or her own payoff, and they negotiate with each other to achieve fairness and address this conflict. However, as we discussed before, some selfish users may cheat their peers and manipulate the system to maximize their own payoffs. Cheat prevention is a critical requirement in many social networks to stimulate user cooperation. It is of ample importance to design monitoring mechanisms to detect and identify misbehaving users and to design cheat-proof cooperation stimulation strategies. Using video fingerprinting as an example, the work in [171] analyzed the complex dynamics among colluders during multiuser collusion, and explored possible monitoring mechanisms to detect and identify misbehaving colluders in multiuser collusion. It considered two types of colluder networks to investigate the impact of network structures on misbehavior detection and identification. One has a centralized structure with a trusted ringleader, while the other is a distributed peer-structured network. The work showed how to accurately identify selfish colluders without falsely accusing others even under attacks. It also evaluated their robustness against framing attacks and quantified the maximum number of framing colluders that they can resist.

The work in [172] considered human factors from a social networking viewpoint, and provided an overview on how to effectively model, analyze, and perform behavior forensics. Various examples of media security and content protection are illustrated in a tutorial manner.

VIII. FINAL THOUGHTS AND FUTURE DIRECTIONS

From the above discussion, the concepts of information forensics have found their way into many applications, ranging from tampering detection, to origin tracking and provenance, to space-time localization. In addressing information forensics questions, the research community so far has mainly relied on empirical testing by individual research groups to demonstrate and compare performances of some specific techniques, often with a small-scale dataset put together in an ad-hoc way. There is little understanding of how to quantitatively address fundamental performance tradeoffs and limits, for no "theory on forensics" has been established yet.

It is essential to establish fundamental theories governing the science of forensics. These include introducing the notion of "forensicability" for devices, channels, and processing systems along an information processing chain. Detection theory, classification theory, and machine learning theory can be utilized to formalize the foundation of forensic analysis for different configurations and combinations of such a chain of processing. Theories on forensicability can provide the community an unprecedented yet very fundamental understanding toward what individual or combinations of device/channel/processing can or cannot be inferred forensically and why, and what fundamental performance limits are.

There is a need to develop a clear definition what forensicability should be and establish a theoretical notion of forensicability from both detection and estimation theory and pattern recognition points of view. Furthermore, there is a need to apply performance analysis to representative forensic scenarios, develop a validation framework, and investigate fundamental performance limits. Such fundamental research will foster systematic development of the young field of information forensics.

As we have seen, to perform forensic analysis, one must start with discovering some traces of evidence. We have reviewed many examples of invisible traces of evidence left in multimedia signals and documents as they go through various operations and devices, and these intrinsic fingerprints have shown strong promise to provide powerful forensic evidence regarding the history and provenance of digital content. But how useful and trustworthy are these forensic traces? Can we measure these quantities and their certainty or uncertainty?

Take component forensics, which was discussed in Section III, as an example. The fact that we can infer the algorithms and their parameter settings inside a device simply from the device outputs also motivates us to ask a series of fundamental questions: Is it possible to extend the developed methodology to perform forensic analysis on other important components as well? If there are multiple components, can we identify them all together? For a chain of processing components inside, are they equally identifiable from the output? Or is there any fundamental identifiability limit due to the order of processing? Can identifying multiple com-

ponents improve the accuracy and confidence in the overall forensic conclusion? If so, how should we perform forensic fusion from the results of multiple components? How much performance improvement can one expect and what is the performance limit if one can perform semi non-intrusive forensics with controlled inputs? Some preliminary effort was made along this line by formulating "forensicability" using parameter estimation and pattern classification theories [175], [176]. Still, there is a very limited theoretical foundation for the emerging field of information forensics to answer many fundamental questions listed above. As such, the research community needs to confront the challenges to build a solid theoretical foundation for the further advancement.

Furthermore, it is necessary to consider all possible uses of forensic technologies. As we have seen in Section III, component forensic techniques can make use of device fingerprints to learn which processing algorithms and parameters are used by digital devices. In addition to serving as useful tools for verifying the authenticity of a signal, these techniques can also potentially be used to reverse engineer a digital device. For the applications that require the protection of proprietary signal processing algorithms inside digital devices, it is critical to develop countermeasures to prevent the undesired use of forensics by other parties.

One potential solution to this problem lies in anti-forensics. Since anti-forensic operations are capable of fooling forensic techniques, it is possible to protect against forensic reverse engineering by integrating anti-forensic operations into a digital device. Proof-of-concept explorations of this idea have been made recently in the case of digital cameras [80], [173]. These are being motivated by component forensic techniques described in Section III that are capable of identifying and estimating the color interpolation algorithms used by different digital cameras. Since such interpolation algorithms are often proprietary, camera manufacturers may wish to prevent other parties from reverse engineering these techniques using component forensics. By incorporating an anti-forensic module into a camera's processing pipeline (including for example, nonlinear filtering, resizing to alter an image's sampling grid, and/or perturbation in a camera's interpolation parameters), component forensic techniques can be potentially subverted, although care must be taken to balance the tradeoff between anti-forensic protection and visual quality.

Additionally, as technology progresses, new digital devices and signal processing techniques continue to emerge. To keep pace with the rapid advancement of technology, it is necessary to identify the forensic fingerprints that these new processing techniques and devices leave behind. For example, compressive sensing has recently emerged as a new method for acquiring and reconstructing sparse signals at rates below the Nyquist rate. Although sparse signals acquired by compressive sensing can in theory be perfectly reconstructed, many signals in practice are not perfectly sparse or are corrupted by noise. The unique distortions introduced into these signals can be used as fingerprints of compressive sensing [174]. Since such intrinsic fingerprints of compressive sensing can be mistaken by existing forensic techniques as that of regular compressions, identifying compressive sensing is a critical step in tracing a signal's processing history [174]. Furthermore, as compressive sensing is integrated into new devices, identifying the use of compressive sensing may become an important step in determining which device captured a digital signal. This is just one example illustrating the importance of identifying forensic traces left by new and emerging technologies.

We also see a recent trend in drawing synergy between intrinsic and extrinsic techniques through building a "forensic hash" [177], [178]. Sending a hash in a file header or through a secondary channel can be viewed as an extrinsic technique in a broad sense, and the hash conventionally provides a binary integrity decision on the data being transmitted. This is no longer sufficient once a data stream may be transcoded in various ways. As reviewed in this paper, while significant progress has been made into developing intrinsic forensic techniques, the computational complexity to address a broad range of forensic questions is still quite high in general. The forensic hash framework aims at utilizing a strategically designed compact string and an associated decision mechanism to go beyond traditional hashes' binary integrity answer. This can help address forensic questions more efficiently and accurately than intrinsic approaches alone. One demonstration in [179] is to construct an alignment component of the hash to facilitate the determination of complex geometric changes between an image under question and the original ground-truth. This approach works hand-in-hand with additional hash components to identify texture or object changes and possibly reconstruct the original version of the image. More generally, the cross fertilization between conventionally distinct classes of solutions may open up new opportunities for advancing information forensics in the years to come.

REFERENCES

[1] G. N. Ali, P.-J. Chiang, A. K. Mikkilineni, J. P. Allebach, G. T. C. Chiu, E. J. Delp, "Intrinsic and extrinsic signatures for information hiding and secure printing with electrophotographic devices," in *Proc. IS&T's NIP19: Int. Conf. Digital Printing Technol.*, vol. 19, pp. 511–515, Sep. 2003.

[2] Z. Wang, M. Wu, W. Trappe, and K. J. R. Liu, "Group-oriented fingerprinting for multimedia forensics," *EURASIP J. Appl. Signal Process.*, vol. 2004, pp. 2153–2173, Oct. 2004.

[3] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection with binary similarity measures," in *Proc. 13th Eur. Signal Process. Conf.*, vol. 1. 2005, pp. 752–755.

[4] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection," *J. Electron. Imag.*, vol. 15, no. 4, pp. 041102-1–041102-17, 2006.

[5] J. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*, 1st ed. Cambridge, U.K.: Cambridge, Cambridge Univ. Press, 2010.

[6] T.-T. Ng, S.-F. Chang, and Q. Sun, "Blind detection of photomontage using higher order statistics," in *Proc. Int. Symp. Circuits Syst.*, vol. 5. May 2004, pp. V–688–V–691.

[7] W. Chen, Y. Q. Shi, and W. Su, "Image splicing detection using 2-D phase congruency and statistical moments of characteristic function," in *Proc. SPIE Security, Steganogr. 9th Watermarking Multimedia Contents*, vol. 6505, 2007, pp. 65050R-1–65050R-8.

[8] Y. Q. Shi, C. Chen, and W. Chen, "A natural image model approach to splicing detection," in *Proc. ACM 9th Workshop Multimedia Security*, 2007, pp. 51–62.

[9] H. Farid and S. Lyu, "Higher-order wavelet statistics and their application to digital forensics," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop*, vol. 8. Jun. 2003, p. 94.

[10] S. Lyu and H. Farid, "How realistic is photorealistic," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 845–850, Feb. 2005.

[11] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui, "Physics-motivated features for distinguishing photographic images and computer graphics," in *Proc. 13th Annu. ACM Int. Conf. Multimedia*, 2005, pp. 239–248.

[12] I. Cox, J. Bloom and M. Miller, *Digital Watermarking: Principles & Practice*, San Mateo, CA, USA: Morgan Kaufman Publishers, 2001.

[13] M. Wu and B. Liu, *Multimedia Data Hiding*. New York, NY, USA: Springer-Verlag, 2003.

[14] M. K. Johnson and H. Farid, "Exposing digital forgeries through chromatic aberration," in *Proc. 8th Workshop Multimedia Security*, 2006, pp. 48–55.

[15] Z. Lin, R. Wang, X. Tang, and H.-Y. Shum, "Detecting doctored images using camera response normality and consistency," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1. Jun. 2005, pp. 1087–1092.

[16] Y.-F. Hsu and S.-F. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2006, pp. 549–552.

[17] Y.-F. Hsu and S.-F. Chang, "Camera response functions for image forensics: An automatic algorithm for splicing detection," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 816–825, Dec. 2010.

[18] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *Proc. 6th Int. Workshop Inf. Hiding*, 2004, pp. 128–147.

[19] B. Mahdian and S. Saic, "Using noise inconsistencies for blind image forensics," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1497–1503, 2009.

[20] X. Pan, X. Zhang, and S. Lyu, "Exposing image forgery with blind noise estimation," in *Proc. 13th ACM Multimedia Workshop Multimedia Security*, 2011, pp. 15–20.

[21] X. Pan, X. Zhang, and S. Lyu, "Exposing image splicing with inconsistent local noise variances," in *Proc. IEEE Int. Conf. Comput. Photogr.*, Apr. 2012, pp. 1–10.

[22] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.

[23] J. Fridrich, D. Soukal, and J. Lukáš, "Detection of copy-move forgery in digital images," in *Proc. Digital Forensic Res. Workshop*, 2003, pp. 1–10.

[24] G. Li, Q. Wu, D. Tu, and S. Sun, "A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 1750–1753.

[25] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Dept. Comput. Sci., Dartmouth College Private Ivy League Res. Univ., Dartmouth, NH, USA, Tech. Rep. TR2004-515, 2004.

[26] W. Luo, J. Huang, and G. Qiu, "Robust detection of region-duplication forgery in digital image," in *Proc. Int. Conf. Pattern Recognit.*, vol. 4. Aug. 2006, pp. 746–749.

[27] S. Bayram, H.T. Sencar, and N. Memon, "An efficient and robust method for detecting copy-move forgery," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 1053–1056.

[28] B. Mahdian and S. Saic, "Detection of copy–move forgery using a method based on blur moment invariants," *Forensic Sci. Int.*, vol. 171, nos. 2–3, pp. 180–189, 2007.

[29] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using sift algorithm," in *Proc. Pacific-Asia Workshop Comput. Intell. Ind. Appl.*, vol. 2. Dec. 2008, pp. 272–276.

[30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[31] X. Pan and S. Lyu, "Region duplication detection using image feature matching," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 857–867, Dec. 2010.

[32] I. Amerini, L. Ballan, R. Caldelli, A. del Bimbo, and G. Serra, "A sift-based forensic method for copy-move attack detection and transformation recovery," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1099–1110, Sep. 2011.

[33] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 758–767, Feb. 2005.

[34] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *Proc. 10th ACM Workshop Multimedia Security*, 2008, pp. 11–20.

[35] M. Kirchner and T. Gloe, "On resampling detection in re-compressed images," in *Proc. Int. Workshop Inf. Forensics Security*, Dec. 2009, pp. 21–25.

[36] A. C. Gallagher, "Detection of linear and cubic interpolation in JPEG compressed images," in *Proc. Can. Conf. Comput. Robot Vis.*, May 2005, pp. 65–72.

[37] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 529–538, Sep. 2008.

[38] M. Stamm and K. J. R. Liu, "Blind forensics of contrast enhancement in digital images," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2008, pp. 3112–3115.

[39] M. C. Stamm and K. J. R. Liu, "Forensic estimation and reconstruction of a contrast enhancement mapping," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Mar. 2010, pp. 1698–1701.

[40] A. Bovik, "Streaking in median filtered images," *IEEE Trans. Acoust., Speech Signal Process.*, vol. 35, no. 4, pp. 493–503, Apr. 1987.

[41] M. Kirchner and J. Fridrich, "On detection of median filtering in digital images," in *Proc. SPIE Electronic Imag., Security, Steganogr., Watermarking Multimedia Contents*, vol. 7541. Feb. 2010, pp. 1–6.

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

[43] T. Pevný, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 215–224, Jun. 2010.

[44] H.-D. Yuan, "Blind forensics of median filtering in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1335–1345, Dec. 2011.

[45] X. Kang, M. C. Stamm, A. Peng, and K. J. R. Liu, "Robust median filtering forensics based on the autoregressive model of median filtered residual," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Dec. 2012, pp. 1–9.

[46] Z. Fan and R. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Trans. Image Process.*, vol. 12, no. 2, pp. 230–235, Feb. 2003.

[47] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. Ray Liu, "Digital image source coder forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 460–475, Sep. 2009.

[48] S. Tjoa, W. S. Lin, H. V. Zhao, and K. J. R. Liu, "Block size forensic analysis in digital images," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, vol. 1. Apr. 2007, pp. 633–636.

[49] S. Tjoa, W. S. Lin, and K. J. R. Liu, "Transform coder classification for digital image forensics," in *Proc. IEEE Int. Conf. Image Process.*, vol. 6. Oct. 2007, pp. 105–108.

[50] J. Lukáš and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Proc. Digital Forensic Res. Workshop*, Aug. 2003, pp. 5–8.

[51] A. C. Popescu and H. Farid, "Statistical tools for digital forensics," in *Proc. 6th Int. Workshop Inf. Hiding*, 2004, pp. 128–147.

[52] T. Pevný and J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 2, pp. 247–258, Jun. 2008.

[53] F. Huang, J. Huang, and Y. Q. Shi, "Detecting double JPEG compression with the same quantization matrix," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 848–856, Dec. 2010.

[54] D. Fu, Y. Q. Shi, and W. Su, "A generalized Benford's law for JPEG coefficients and its applications in image forensics," in *Proc. SPIE, Electron. Imag., Security, Steganogr., Watermarking Multimedia Contents*, vol. 6505. Jan. 2007, pp. 1–11.

[55] B. Li, Y. Q. Shi, and J. Huang, "Detecting doubly compressed JPEG images by using mode based first digit features," in *Proc. IEEE Workshop Multimedia Signal Process.*, Oct. 2008, pp. 730–735.

[56] J. He, Z. Lin, L. Wang, and X. Tang, "Detecting doctored JPEG images via DCT coefficient analysis," in *Proc. Comput. Vis.ŰECCV*, vol. 3953. Mar. 2006, pp. 423–435.

[57] Z. Lin, J. He, X. Tang, and C.-K. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognit.*, vol. 42, no. 11, pp. 2492–2501, 2009.

[58] Y.-L. Chen and C.-T. Hsu, "Detecting recompression of JPEG images via periodicity analysis of compression artifacts for tampering detection," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 2, pp. 396–406, Jun. 2011.

[59] T. Bianchi and A. Piva, "Detection of nonaligned double JPEG compression based on integer periodicity maps," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 842–848, Apr. 2012.

[60] T. Bianchi and A. Piva, "Image forgery localization via block-grained analysis of JPEG artifacts," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1003–1017, Jun. 2012.

[61] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 154–160, Mar. 2009.

[62] W. Li, Y. Yuan, and N. Yu, "Passive detection of doctored JPEG image via block artifact grid extraction," *Signal Process.*, vol. 89, no. 9, pp. 1821–1829, 2009.

[63] S. Ye, Q. Sun, and E.-C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 12–15.

[64] M. Barni, A. Costanzo, and L. Sabatini, "Identification of cut & paste tampering by means of double-JPEG detection and image segmentation," in *Proc. IEEE Int. Symp. Circuits Syst.*, Jun. 2010, pp. 1687–1690.

[65] W. Luo, Z. Qu, J. Huang, and G. Qiu, "A novel method for detecting cropped and recompressed image block," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, vol. 2. Apr. 2007, pp. 217–220.

[66] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double quantization," in *Proc. 11th ACM Workshop Multimedia Security*, 2009, pp. 39–48.

[67] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double MPEG compression," in *Proc. 8th Workshop Multimedia Security*, 2006, pp. 37–47.

[68] M. C. Stamm, W. S. Lin, and K. J. R. Liu, "Temporal forensics and anti-forensics for motion compensated video," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 4, pp. 1315–1329, Aug. 2012.

[69] M. K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *Proc. 7th Workshop Multimedia Security*, 2005, pp. 1–10.

[70] M. K. Johnson and H. Farid, "Exposing digital forgeries in complex lighting environments," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 450–461, Sep. 2007.

[71] M. K. Johnson and H. Farid, "Exposing digital forgeries through specular highlights on the eye," in *Proc. Inf. Hiding*, vol. 4567. 2007, pp. 311–325.

[72] V. Conotter, J. F. O'Brien, and H. Farid, "Exposing digital forgeries in ballistic motion," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 1, pp. 283–296, Feb. 2012.

[73] 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.

[74] A. Swaminathan, M. Wu, and K. J. Ray Liu, "Non-intrusive forensic analysis of visual sensors using output images," in *Proc. IEEE Conf. Acoustic, Speech Signal Process.*, May 2006, pp. 1–5.

[75] S. Kawamura, "Capturing images with digital still cameras," *IEEE Micro*, vol. 18, no. 6, pp. 14–19, Nov.–Dec. 1998.

[76] J. Adams, "Interaction between color plane interpolation and other image processing functions in electronic photography," in *Proc. SPIE, Cameras Syst. Electron. Photogr. Sci. Imag.*, vol. 2416. Feb. 1995, pp. 144–151.

[77] J. Adams, K. Parulski, and K. Spaulding, "Color processing in digital cameras," *Proc. IEEE*, vol. 18, no. 6, pp. 20–30, Nov.–Dec. 1998.

[78] A. C. Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," *IEEE Trans. Signal Process.*, vol. 53, no. 10, pp. 3948–3959, Oct. 2005.

[79] A. Swaminathan, M. Wu, and K. J. Ray Liu, "Component forensics of digital cameras: A non-intrusive approach," in *Proc. Conf. Inf. Sci. Syst.*, Mar. 2006, pp. 1–6.

[80] W.-H. Chuang and M. Wu, "Robustness of color interpolation identification against anti-forensic operations," in *Proc. Inf. Hiding Workshop*, May 2012, pp. 16–30.

[81] H. Cao and A. C. Kot, "Accurate detection of demosaicing regularity for digital image forensics," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 4, pp. 899–910, Dec. 2009.

[82] S. Bayram, H. T. Sencar, N. Memon, and I. Avcibas, "Source camera identification based on CFA interpolation," in *Proc. IEEE Int. Conf. Image Process.*, vol. 3. Sep. 2005, pp. 69–72.

[83] S. Bayram, H. T. Sencar, and N. Memon, "Improvements on source camera-model identification based on CFA interpolation," in *Proc. Int. Conf. Digital Forensics*, Jan. 2006, pp. 1–9.

[84] J. Lukáš, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.

[85] J. Fridrich, "Digital image forensics using sensor noise," *IEEE Signal Proc. Mag.*, vol. 26, no. 2, pp. 26–37, Mar. 2009.

[86] C.Ű.T. Li, "Source camera identification using enhanced sensor pattern noise," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 2, pp. 280–287, Jun. 2010.

[87] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Source digital camcorder identification using sensor photo-response nonuniformity," *Proc. SPIE Int. Soc. Opt. Eng.*, vol. 6505, p. 65051G, Jan. 2007.

[88] S. McCloskey, "Confidence weighting for sensor fingerprinting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2008, pp. 1–6.

[89] W.-H. Chuang, H. Su, and M. Wu, "Exploring compression effects for improved source camera identification using strongly compressed video," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 1953–1956.

[90] T.-T. Ng and M.-P. Tsui, "Camera response function signature for digital forensicsŰPart I: Theory and data selection and Part II: Signature extraction," in *Proc. IEEE Workshop Inf. Forensics Security*, Dec. 2009, pp. 156–160.

[91] H. Gou, A. Swaminathan, and M. Wu, "Robust scanner identification based on noise features," in *Proc. SPIE Electron. Imag. Conf. Security, Watermarking Steganography*, vol. 6505. 2007, pp. 1–11.

[92] N. Khanna, A. K. Mikkilineni, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Scanner identification using sensor pattern noise," in *Proc. SPIE Electron. Imag. Conf. Security, Watermarking Steganography*, vol. 6505. 2007, pp. 1–11.

[93] N. Khanna, A. K. Mikkilineni, and E. J. Delp, "Scanner identification using feature-based processing and analysis," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 1, pp. 123–139, Mar. 2009.

[94] H. Gou, A. Swaminathan, and M. Wu, "Intrinsic sensor noise features for forensic analysis on scanners and scanned images," *IEEE Trans. Info. Forensics Security*, vol. 4, no. 3, pp. 476–491, Sep. 2009.

[95] N. Khanna, G. T.-C. Chiu, J. P. Allebach, E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2008, pp. 1653–1656.

[96] C.E. McKay, A. Swaminathan, H. Gou, and M. Wu, "Image acquisition forensics: Forensic analysis to identify imaging source," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2008, pp. 1657–1660.

[97] A. C. Gallagher and T. Chen, "Image authentication by detecting traces of demosaicing," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2008, pp. 1–8.

[98] A. E. Dirik and N. Memon, "Image tamper detection based on demosaicing artifacts," in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 1497–1500.

[99] A. Swaminathan, M. Wu, and K. J. R. Liu, "Image tampering identification using blind deconvolution," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 2309–2312.

[100] A. Swaminathan, M. Wu, and K. J. R. Liu, "Digital image forensics via intrinsic fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 101–117, Mar. 2008.

[101] W.-H. Chuang, A. Swaminathan, and M. Wu, "Tampering identification using empirical frequency response," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2009 pp. 1517–1520.

[102] J. Lukáš, J. Fridrich, and M. Goljan, "Detecting digital image forgeries using sensor pattern noise," in *Proc. SPIE Electron. Imag.*, vol. 6072. 2006, p. 60720Y.

[103] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.

[104] H. Gou, A. Swaminathan, and M. Wu, "Noise features for image tampering detection and steganalysis," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2007, pp. 97–100.

[105] M. Bollen and I. Gu, *Signal Processing of Power Quality Disturbances*. New York, NY, USA: Wiley, 2006.

[106] NPR, (2009). *Visualizing the U.S. Electric Grid* [Online]. Available: http://www.npr.org/templates/story/ story.php?storyId=110997398

[107] R. W. Sanders, "Digital authenticity using the electric network frequency," in *Proc. 33rd AES Int. Conf. Audio Forensics, Theory Practice*, Jun. 2008, pp. 1–6.

[108] C. Grigoras, "Applications of ENF criterion in forensics: Audio, video, computer and telecommunication analysis," *Forensic Sci. Int.*, vol. 167, nos. 2–3, pp. 136–145, 2007.

[109] R. Garg, A. L. Varna, and M. Wu, "Seeing' ENF: Natural time stamp for digital video via optical sensing and signal processing," in *Proc. ACM Multimedia*, Nov. 2011, pp. 1–10.

[110] O. Ojowu, J. Karlsson, J. Li, and Y. Liu, "ENF extraction from digital recordings using adaptive techniques and frequency tracking," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 4, pp. 1330–1338, Aug. 2012.

[111] A. Hajj-Ahmad, R. Garg, and M. Wu, "Instantaneous frequency estimation and localization for ENF signals," in *Proc. Annu. Summit Conf. Asia-Pacific Signal Inf. Process. Assoc.*, Dec. 2012. pp. 1–10.

[112] European Network of Forensic Science Institutes (ENFSI), "Best Practice Guidelines for ENF Analysis in Forensic Authentication of Digital Evidence," by ENFSI Forensic Speech and Audio Analysis Working Group, Document# FSAAWG-BPM-ENF-001, Jun. 2009.

[113] R. Garg, A. Hajj-Ahmad, and M. Wu, "Geo-location estimation from electrical network frequency signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2013, pp. 1–3.

[114] D. Rodriguez, J. Apolinario, and L. Biscainho, "Audio authenticity: Detecting ENF discontinuity with high precision phase analysis," *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 534–543, Sep. 2010.

[115] T. Gloe, M. Kirchner, A. Winkler, and R. Böhme, "Can we trust digital image forensics," in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 78–86.

[116] M. Kirchner and R. Böhme, "Hiding traces of resampling in digital images," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 4, pp. 582–592, Dec. 2008.

[117] M. C. Stamm and K. J. R. Liu, "Anti-forensics of digital image compression," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 3, pp. 1050–1065, Sep. 2011.

[118] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Anti-forensics of JPEG compression," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, Mar. 2010, pp. 1694–1697.

[119] M. C. Stamm and K. J. R. Liu, "Wavelet-based image compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 1737–1740.

[120] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Undetectable image tampering through JPEG compression anti-forensics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 2109–2112.

[121] X. Chu, M. C. Stamm, Y. Chen, and K. J. R. Liu, "Concealability-rate-distortion tradedoff in image compression anti-forensics," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2013, pp. 1–3.

[122] M. C. Stamm and K. J. R. Liu, "Anti-forensics for frame deletion/addition in MPEG video," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2011, pp. 1876–1879.

[123] M. Goljan, J. Fridrich, and M. Chen, "Defending against fingerprint-copy attack in sensor-based camera identification," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 1, pp. 227–236, Mar. 2011.

[124] S. Lai and R. Böhme, "Anti-forensics for frame deletion/addition in MPEG video," in *Proc. Inf. Hiding Conf.*, May 2011, pp. 1876–1879.

[125] G. Valenzise, V. Nobile, M. Tagliasacchi, and S. Tubaro, "Forensics versus. anti-forensics: A decision and game theoretic framework," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Sep. 2011, pp. 1749–1752.

[126] W. H. Chuang, R. Garg, and M. Wu, "How secure are power network signature based time stamps," in *Proc. ACM Conf. Comput. Commun. Security*, Oct. 2012, pp. 428–438.

[127] M. Barni and B. Tondi, "The source identification game: An information-theoretic perspective," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 3, pp. 450–463, Mar. 2013.

[128] K. J. R. Liu, W. Trappe, Z. J. Wang, M. Wu, and H. Zhao, *Multimedia Fingerprinting Forensics for Traitor Tracing*, Nasr, Cairo, Egypt, Hindawi Publishing Corporations, vol. 4. Dec. 2005.

[129] M. Wu, W. Trappe, Z. Wang, and K. J. R. Liu, "Collusion resistant fingerprinting for multimedia," *IEEE Signal Process. Mag.*, vol. 21, no. 2, pp. 15–27, Mar. 2004.

[130] M. Wu and Q. Sun, "Video security and protection," in *The Essential Guide to Video Processing*. New York, NY, USA: Elsevier, 2009.

[131] I. Cox, J. Kilian, F. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Proc.*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[132] C. Podilchuk and W. Zeng, "Image adaptive watermarking using visual models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 525–539, May 1998.

[133] C.-Y. Lin, M. Wu, Y.-M. Lui, J. A. Bloom, M. L. Miller, and I. J. Cox, "Rotation, scale, and translation resilient public watermarking for images," *IEEE Trans. Image Process.*, vol. 10, no. 5, pp. 767–782, May 2001.

[134] J. Lubin, J. Bloom, and H. Cheng, "Robust, content-dependent, high-fidelity watermark for tracking in digital cinema," in *Proc. Security Watermarking Multimedia Contents, SPIE*, vol. 5020, Jan. 2003, pp. 536–545.

[135] B. Chen and G. W. Wornell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding," *IEEE Trans. Inf. Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.

[136] P. Moulin and J. A. O'Sullivan, "Information-theoretic analysis of information hiding," *IEEE Trans. Inf. Theory*, vol. 49, no. 3, pp. 563–593, Mar. 2003.

[137] J. G. Proakis, *Digital Communications*, 4th ed. New York, NY, USA: McGraw-Hill, 2000.

[138] H. Gou and M. Wu, "Data hiding in curves with applications to fingerprinting maps," *IEEE Trans. Signal Proc.*, vol. 53, no. 10, pp. 3988–4005, Oct. 2005.

[139] F. Ergun, J. Kilian, and R. Kumar, "A note on the limits of collusion-resistant watermarks," *Eurocrypt Lecture Notes Comput. Sci.*, vol. 1592, pp. 140–149, Jan. 1999.

[140] J. Kilian, T. Leighton, L. Matheson, T. Shamoon, R. Tarjan, and F. Zane, "Resistance of digital watermarks to collusive attacks," in *Proc. IEEE Int. Symp. Inf. Theory*, Aug. 1998, p. 271.

[141] Z. J. Wang, M. Wu, H. Zhao, W. Trappe, and K. J. R. Liu, "Resistance of orthogonal gaussian fingerprints to collusion attacks," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2003, pp. 617–620.

[142] Z. J. Wang, M. Wu, H. Zhao, W. Trappe, and K. J. R. Liu, "Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation," *IEEE Trans. Image Proc.*, vol. 14, no. 6, pp. 804–821, Jun. 2005.

[143] N. Kiyavash and P. Moulin, "Performance of orthogonal fingerprinting codes under worst-case noise," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 293–301, Sep. 2009.

[144] W. Trappe, M. Wu, and K. J. R. Liu, "Collusion-resistant fingerprinting for multimedia," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2002, pp. 3309–3312.

[145] S. He and M. Wu, "Joint coding and embedding techniques for multimedia fingerprinting," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 2, pp. 231–247, Jun. 2006.

[146] S. He and M. Wu, "Performance study of ecc-based collusion-resistant multimedia fingerprinting," in *Proc. 38th CISS*, Mar. 2004, pp. 827–832.

[147] H. S. Stone, "Analysis of attacks on image watermarks with randomized coefficients," NEC Res. Inst. Princeton Univ., Princeton, NJ, USA, Tech. Rep. 96-045, 1996.

[148] H. Zhao, M. Wu, Z. J. Wang, and K. J. R. Liu, "Nonlinear collusion attacks on independent fingerprints for multimedia," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2003, pp. 613–616.

[149] H. Zhao, M. Wu, Z. J. Wang, and K. J. R. Liu, "Forensic analysis of nonlinear collusion attacks for multimedia fingerprinting," *IEEE Trans. Image Proc.*, vol. 14, no. 5, pp. 646–661, May 2005.

[150] W. Trappe, M. Wu, Z. J. Wang, and K. J. R. Liu, "Anti-collusion fingerprinting for multimedia," *IEEE Trans. Signal Process.*, vol. 51, no. 4, pp. 1069–1087, Apr. 2003.

[151] K. Su, D. Kundur, and D. Hatzinakos, "Statistical invisibility for collusion-resistant digital video watermarking," *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 43–51, Feb. 2005.

[152] M. D. Swanson, B. Zhu, and A. T. Tewfik, "Multiresolution scene-based video watermarking using perceptual models," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 540–550, May 1998.

[153] S. He and M. Wu, "Collusion-resistant video fingerprinting for large user group," *IEEE Trans. Info. Forensics Security*, vol. 2, no. 4, pp. 697–709, Dec. 2007.

[154] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Trans. Inf. Theory*, vol. 44, no. 5, pp. 1897–1905, Sep. 1998.

[155] Y. Yacobi, "Improved Boneh-Shaw content fingerprinting," in *Proc. CT-RSA Lecture Notes Comput. Sci.*, vol. 2020, pp. 378–91, 2001.

[156] G. Tardos, "Optimal probabilistic fingerprint codes," in *Proc. 35th Annu. ACM Symp. Theory Comput.*, 2003, pp. 116–125.

[157] T. Furon, A. Guyader, and F. Cerou, "On the design and optimization of tardos probabilistic fingerprinting codes," in *Proc. Inf. Hiding Workshop*, vol. 5284, 2008, pp. 341-Ű356.

[158] J. Dittmann, P. Schmitt, E. Saar, J. Schwenk, and J. Ueberberg, "Combining digital watermarks and collusion secure fingerprints for digital images,"*J. Electron. Imag.*, vol. 9, pp. 456–467, Jan. 2000.

[159] A. Barg, G. R. Blakley, and G. Kabatiansky, "Digital fingerprinting codes: Problem statements, constructions, identification of traitors," *IEEE Trans. Inform. Theory*, vol. 49, no. 4, pp. 852Ű-865, Apr. 2003.

[160] R. Safavi-Naini and Y. Wang, "Collusion secure q-ary fingerprinting for perceptual content," *Security Privacy Digital Rights Manage.*, vol. 2320, pp. 57–75, Feb. 2002.

[161] F. Zane, "Efficient watermark detection and collusion security," in *Proc. 4th Int. Conf. Financial Cryptogr.*, Feb. 2000, pp. 21–32.

[162] J. N. Staddon, D. R. Stinson, and R. Wei, "Combinatorial properties of frameproof and traceability codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 3, pp. 1042–1049, Mar. 2001.

[163] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. Upper Saddle River, NJ, USA: Prentice Hall, 1995.

[164] H. V. Zhao and K. J. R. Liu, "Fingerprint multicast for secure video streaming," *IEEE Trans. Image Process.*, vol. 15, no. 1, pp. 12–29, Jan. 2006.

[165] H. V. Zhao and K. J. R. Liu, "Behavior forensics for scalable multiuser collusion: Fairness versus effectiveness," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 3, pp. 311–329, Sep. 2006.

[166] W. S. Lin, H. V. Zhao, and K. J. R. Liu, "Multi-user collusion behavior forensics: Game-theoretic formulation of fairness dynamics," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2007, pp. 109–112.

[167] W. S. Lin, H. V. Zhao, and K. J. R. Liu, "Game-theoretic strategies and equilibriums in multimedia fingerprinting social networks," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 191–205, Apr. 2011.

[168] G. Owen, *Game Theory*, 3rd ed. San Diego, CA, USA: Academic Press, 1995.

[169] H. V. Zhao and K. J. R. Liu, "Traitor-within-traitor behavior forensics: Strategy and risk minimization," *IEEE Trans. Inf. Forensics Security*, vol. 1, no. 4, pp. 440–456, Dec. 2006.

[170] W. S. Lin, H. V. Zhao, and K. J. R. Liu, "Behavior forensics with side information for multimedia fingerprinting social networks," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 4, pp. 911–927, Dec. 2009.

[171] H. V. Zhao and K. J. R. Liu, "Impact of social network structure on multimedia fingerprinting misbehavior detection and identification," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 4, pp. 687–703, Aug. 2010.

[172] H. V. Zhao, W. S. Lin, and K. J. R. Liu, "Behavior modeling and forensics for multimedia social networks," *IEEE Signal Process. Mag.*, vol. 26, no. 1, pp. 118–139, Jan. 2009.

[173] M. C. Stamm, W. S. Lin, and K. J. R. Liu, "Protection against reverse engineering in digital cameras," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 1–11.

[174] X. Chu, M. C. Stamm, and K. J. R. Liu, "Forensic identification of compressive sensing in nearly sparse signals," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2012, pp. 257–260.

[175] A. Swaminathan, M. Wu, and K. J. R. Liu, "A component estimation framework for information forensics," in *Proc. IEEE Workshop Multimedia Signal Process.*, Oct. 2007, pp. 397–400.

[176] A. Swaminathan, M. Wu, and K. J. R. Liu, "A pattern classification framework for theoretical analysis of component forensics," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2008, pp. 1–3.

[177] W. Lu, A. L. Varna, and M. Wu, "Forensic hash for multimedia information," *Proc. SPIE*, vol. 7541, pp. 7541–7544, Jan. 2010.

[178] W. Lu and M. Wu, "Multimedia forensic hash based on visual words," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2010, pp. 989–992.

[179] W. Lu and M. Wu, "Seam carving estimation using forensic hash," in *Prof. ACM Multimedia Security Workshop*, Sep. 2011, pp. 1–11.

MATTHEW C. STAMM (S'08–M'12) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, MD, USA, in 2004, 2011, and 2012, respectively. He is currently a Post-Doctoral Research Associate with the Department of Electrical and Computer Engineering, University of Maryland. His research interests include signal processing and information security with a focus on digital multimedia forensics and anti-forensics.

Dr. Stamm received the Dean's Doctoral Research Award in 2012 from the A. James Clark School of Engineering at the University of Maryland. Additionally, he received a Distinguished Teaching Assistant Award in 2006, a Future Faculty Fellowship in 2010, and the Ann G. Wylie Fellowship in 2011 from the University of Maryland. From 2004 to 2006, he was a Radar Systems Engineer with the Johns Hopkins University Applied Physics Laboratory.

MIN WU (S'95–M'01–SM'06–F'11) received the B.E. degree in electrical engineering and the B.A. degree in economics from Tsinghua University, Beijing, China (both with the highest honors), in 1996, and the Ph.D. degree in electrical engineering from Princeton University, Princeton, NJ, USA, in 2001. Since 2001, she has been with University of Maryland, College Park, MD, USA, where she is currently a Professor and a University Distinguished Scholar-Teacher. She leads the Media and Security Team at the University of Maryland. Her current research interests include information security and forensics and multimedia signal processing. She has co-authored two books and holds eight U.S. patents on multimedia security and communications.

Dr. Wu is a co-recipient of two Best Paper Awards from the IEEE Signal Processing Society and EURASIP, respectively. She received a NSF CAREER Award in 2002, a TR100 Young Innovator Award from the MIT Technology Review Magazine in 2004, an ONR Young Investigator Award in 2005, a Computer World "40 Under 40" IT Innovator Award in 2007, an IEEE Mac Van Valkenburg Early Career Teaching Award in 2009, and a University of Maryland Invention of the Year Award in 2012. She has served as Vice President–Finance of the IEEE Signal Processing Society from 2010 to 2012 and Chair of the IEEE Technical Committee on Information Forensics and Security from 2012 to 2013. She is an IEEE fellow for contributions to multimedia security and forensics.

K. J. RAY LIU (F'03) was named a Distinguished Scholar-Teacher of University of Maryland, College Park, MD, USA, in 2007, where he is a Christine Kim Eminent Professor of Information Technology. He leads the Maryland Signals and Information Group conducting research encompassing broad areas of signal processing and communications with recent focus on cooperative and cognitive communications, social learning and network science, information forensics and security, and green information and communications technology.

He is a recipient of numerous honors and awards, including the IEEE Signal Processing Society Technical Achievement Award and Distinguished Lecturer. He also received various teaching and research recognitions from University of Maryland including University-Level Invention of the Year Award; and Poole and Kent Senior Faculty Teaching Award, Outstanding Faculty Research Award, and Outstanding Faculty Service Award, all from A. James Clark School of Engineering. He is a fellow of AAAS.

Dr. Liu is President of the IEEE Signal Processing Society where he has served as Vice President - Publications and Board of Governor. He was the Editor-in-Chief of the *IEEE Signal Processing Magazine* and the founding Editor-in-Chief of *EURASIP Journal on Advances in Signal Processing*.