REVERSE

REVerse engineering of audio-VIsual coNtent Data

Grant Agreement No. 268478

Deliverable D3.1
State-of-the-art on multimedia footprint detection

Lead partner for this deliverable: Imperial
Version: 1.0

Dissemination level: Public

September 26, 2011

Contents

Chapter 1

Introduction

With the rapid proliferation of inexpensive hardware devices that enable the acquisition of audiovisual data, new types of multimedia digital objects (audio, images and videos) can be readily created, stored, transmitted, modified and tampered with. Nowadays, the duplication of digital objects is a quite straightforward procedure and the storage of copies on reliable physical devices has become rather inexpensive. As a consequence, during its lifetime, a multimedia object might go through several processing stages, including multiple analog-to-digital (A/D) and digital-to-analog (D/A) conversions, coding and decoding, transmission, editing (aimed at enhancing the quality, creating new content by mixing pre-existing material, or tampering with the content).

These facts highlight the need for methods and tools that enable the reconstruction of the complete history of a digital object in order to assess its authenticity or its quality, and to facilitate the indexing of different versions of the same multimedia object.

The history of multimedia objects can be described in terms of complex information processing chains, whereby each processing operator alters the underlying features of the content in a characteristic and detectable manner. Footprint detection works by finding the traces that are left when a digital object goes through various processing blocks in the processing chain; when the processing parameters are not known, they can be estimated by analyzing the corresponding footprints. The estimate is then used by the footprint detector.

The aim of this report is to provide a comprehensive overview of the state-of-the-art in multimedia footprint detection and footprint parameter estimation. Several criteria could be used for organizing this overview: a historical sequence would better illustrate the evolution of the research field and the shifts in focus over time; a classification upon the major techniques on which the works rely (e.g., information theoretic vs. signal processing based, or deterministic vs. probabilistic) would serve to pinpoint the main research venues thus far. However, for the sake of clarity, we have decided to divide the review in three chapters, each related to one processing block: Chapter 2 to acquisition, Chapter 3 to coding, and Chapter 4 to editing. Each chapter is further divided into three main sections, each focusing on the signal modality of interest (i.e., image, video, audio).

The above sections are made as self-contained as possible, notwithstanding the fact that footprint detection normally requires the joint analysis of different processing stages. For example, the presence of (malicious) editing is normally detected by finding acquisition or coding footprints; in particular, the presence of double compression or double acquisition is an indication that a multimedia object has undergone some editing. The review will highlight such connections when appropriate and will refer to the relevant sections for further details.

We briefly summarize below some of the main findings of our review. The interested reader will find

all the necessary details in the corresponding sections of the review.
Image and video acquisition footprints arise from the overall combination of individual traces left by each single stage in the acquisition process cascade. Acquisition fingerprint detection methods found in the literature are characterised by high success rates; however, they normally require images captured under controlled conditions or a multitude of images available for a single device. This is not always possible, especially taking into account low-cost devices with high noise components.
Significantly, limited attention has been devoted to characterisation of fingerprints arising from chains of acquisition stages, even though the few methods that considered simultaneously more than one processing stage enjoyed increased classification performance [4], [2]. This would suggest that focus on the complete acquisition system would be desirable for the realisation of practical algorithms.
After acquisition, multimedia object are typically lossy compressed in order to save storage and network resources. Lossy compression inevitably leaves itself characteristic footprints, which are related to the specific coding architecture. Most of the literature has focused on studying the processing history of JPEG-compressed images, proposing methods to: i) detect whether an image was JPEG-compressed; ii) determine the quantization parameters used; iii) reveal traces of double JPEG compression. JPEG compression belongs to the broader family of block-based image coding schemes. As such, several works have targeted methods to detect footprints related to blocking artifacts. The aforementioned approaches assume the viewpoint of the analyst who is interested in determining the processing history. Recently, some works have taken the perspective of a knowledgeable adversary, whose goal is to deceive footprint detection by performing ad-hoc anti-forensics processing.
Similarly to images, video sequences are lossy compressed. Several coding standards have been defined over the years by international standardization bodies, notably ITU-T and MPEG. Although such standards share a common hybrid DCT-DPCM coding architecture, each encoder is characterized by specific coding tools, thus leading to a large number of coding configurations that need to be considered. Due to the inherent complexity of the problem, understanding the coding history of video sequences is still in its infancy. Just a few works have addressed the problem of estimating the coding parameters, e.g. quantization parameters, coding modes, motion vectors, and detect double video compression, mostly for the case of MPEG-2 and H.264/AVC video. Video is typically transmitted over error prone networks. In case of packet losses, the decoder applies error concealment methods to improve the perceptual quality of the received signal. Error concealment is bound to leave footprints which can be exploited to reveal the characteristics of the network.
In much the same way as for acquisition and compression, the basic idea underlying editing detection is that each processing leaves some traces hidden into the media. These traces can be searched either at a "statistical level", by analyzing the media in some proper domain, or at the "scene level", for example by looking for inconsistencies in shadows or lighting. Furthermore, as already highlighted before, many editing techniques try to infer tampering by detecting double compression or double acquisition. Most of the work on editing has focused on images and in part audio. Much less work has been developed targeting video, probably due to the huge amount of data concerned.
In particular, for audio content, most of the existing approaches are motivated by audio forensic research. Traces of the electric network frequency embedded in audio recordings may enable a unique determination of the acquisition time. In addition, discontinuities of the electric network frequency can be used to detects edits such as removal, duplication or splicing of audio segments. Other methods to detect such edits are based on time- or frequency-domain properties of the signal. Modifications by signal processing techniques, such as filtering, mixing, or the application of nonlinear effects, forms another class of operations. Several approaches to detect such modifications are reported. While the characterization of the recording environment gained only limited interest for footprint detection so far, there exists profound knowledge from research areas such as blind

dereverberation which is likely to be applicable to this problem.
To conclude, we notice from the above summary as well as from the complete survey of the state-of-the-art in the following chapters that most of the past work has focused on detection and/or parameter estimation of footprints left in still images, while research for video and audio is still in its infancy. Moreover, most of previous activities have focused on single signal modalities and on processing operators of the same kind. All this further highlight the importance and value of the research activity to be undertaken by the REWIND consortium.

Chapter 2

Acquisition

2.1 Image Acquisition

Acquisition-based footprints on still images can be studied from different perspectives. On the one hand, much of the research efforts have been focused on characterizing particular stages during the camera acquisition process for device identification, forgery detection or device linking purposes. On the other hand, image acquisition is also performed with digital scanners, and many of the techniques developed for camera footprint analysis have been translated to their scanner equivalents. Finally, rendering of photorealistic computer graphics (PRCG) requires application of a physical light transport and camera acquisition models, and can be thought of as a third acquisition modality. For digital camera image acquistion, the process can be summarized by the stages shown schematically below:

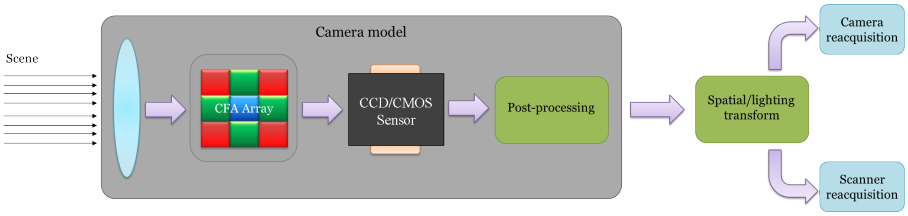

Figure 2.1: Illustration of the image acquisition process.

From the diagram above, the target scene will first be distorted by the capturing lense, before being mosaiced by an RGB Colour Filter Array (CFA). Pixel values are then stored on the internal CCD/CMOS array, and then post-processed for software-based gamma correction, edge enhancement and often JPEG compression. The captured image is then either displayed/projected on screen or printed and can then be recaptured either with a second camera setup or a digital scanner. In this case, geometric distortions due to the orientation of the flat photograph with respect to the second camera as well as the lighting source in the reacquisition setup will transform the recaptured image. While each of the stages above leaves a characteristic footprint on the captured image, so far each processing block has been considered in isolation, studying the digital footprints left regardless of the remaining processing stages. This is certainly useful as an initial study of the individual camera footprints that can be found within a digital image. However, it leaves scope for analysis of operator chains. To further corroborate this idea, several methods have been presented where cues from more

than one stage are simultaneously taken into account, albeit based on either heuristics or black-box classifiers, rather than on a formal understanding of cascading operators. This approach has been proven to boost the accuracy of device identification algorithms [2] [3] [4].
In the following sections the state-of-the-art concerning digital footprints left by individual operators will be presented, followed by the work on scanned image analysis and PRCG image detection. A comprehensive survey on non-intrusive footprint detection methods was also presented in [5].

2.1.1 PRNU-based footprints

Each image acquired with a given camera presents a Photo Response Non-Uniformity (PRNU) noise. This is due to a combination of factors including imperfections during the CCD/CMOS manufacturing process, silicone inhomogeneities and thermal noise. PRNU is a high-frequency multiplicative noise that is unique to each camera. However, it is generally stable throughout the camera's lifetime in normal operating conditions and is correlated with cameras of the same brand. This makes it ideal not just for device identification, but also for device linking and, if inconsistencies in the PRNU within the image are found in certain areas, for forgery detection.
In its general form shared by most works in the area, a simplified model for the image signal is assumed in order to develop low-complexity algorithms that would be applicable for most camera models and brands. In these cases, the sensor output is expressed as:

$$\mathbf{I} = g^{\gamma}[(\mathbf{1}+\mathbf{K})\mathbf{Y} + \Lambda]^{\gamma} + \Theta_q \tag{2.1}$$

Where $\mathbf{I}$ is the signal in a selected colour channel, $\mathbf{Y}$ is the incident light intensity, g is the colour channel gain and γ the gamma correction factor, while $\mathbf{K}$ is a zero-mean noise-like signal responsible for PRNU, Λ is the combination of other internal noise sources and Θ_q is the quantisation noise.
Given that in natural images the dominant term of the equation will be the incident light intensity, the $\mathbf{Y}$ can be factored out and after truncation of the Taylor expansion a simplified model can be expressed as:

$$\mathbf{I} = \mathbf{I}^{(0)} + \mathbf{K}\mathbf{I}^{(0)} + \Psi \tag{2.2}$$

where $\mathbf{I}^{(0)} = (g\mathbf{Y})^{\gamma}$ is the captured light in absence of noise, $\mathbf{I}^{(0)}K$ is the PRNU term, and Ψ is a combination of random noise components. The PRNU term is then normally estimated by taking N images of smooth, bright (but not saturated) areas which are then denoised and used for host signal rejection and suppression of the noiseless term:

$$\mathbf{W} = I - \mathbf{I}^{(0)} = \mathbf{I}\mathbf{K} + \Phi \tag{2.3}$$

where Φ is the sum of Ψ and two additional terms introduced by the denoising filter. The maximum likelihood predictor for $\mathbf{K}$ is then formulated as [6]:

$$\mathbf{K} = \frac{\sum_{k=1}^{N} \mathbf{W}_k \mathbf{I}_k}{\sum_{k=1}^{N} (\mathbf{I}_k)^2} \tag{2.4}$$

Most of the work in this area focuses on making the PRNU estimation more robust, as its reliability is linked to the presence of bright, low-frequency homogeneous areas in the image. In [6], controlled camera-specific training data is used to obtain a maximum likelihood predictor for the PRNU. Its robustness is improved in [7], where image and PRNU averaging is employed. The algorithm is also tested in more realistic settings. In [8] the PRNU is estimated exclusively based on regions of high SNR between estimated PRNU and total noise residual to minimize the impact of high

frequency image regions. Similarly, in [9] the authors propose a scheme that attenuates strong PRNU components which are likely to have been affected by high frequency image components. In [10], a combination of features from the extracted footprint, including block covariance and image moments, are used for camera classification purposes.
In [11] the problem of complexity is investigated, since the complexity of footprint detection is proportional to number of pixels in the image. The authors developed "digests" which allow for fast search algorithms to take place within large image databases. In [12] PRNUs from the same camera are clustered from large databases and the newly clustered images are used to classify additional entries. The method was also tested for robustness in case of JPEG compression.
Robustness is further investigated in [13], where the task of PRNU identification after attacks of a non-technical user is tested. Denoising, demosaicing, and recompression operations are taken into account. Finally, in [4] noise from the Color Filter Array (CFA) is decoupled from PRNU leading to increased classification performance.

2.1.2 Camera identification from CFA patterns

Excluding professional triple-CCD/CMOS cameras, the vast majority of consumer cameras acquire a single color per pixel. The sensor array is arranged in the form of a Bayer array for the RGB components. A direct consequence of this physical configuration is that one third of the image is sensed directly, while the rest is interpolated from the Bayer array. This introduces specific correlations in the image spectrum. While the spectrum is not unique to a single camera, thus it is not as discriminative as the PRNU information, CFA pattern information can still be used to show that a given image was not taken with a given camera.
In [14], seven different interpolation algorithms were studied. An Expectation-Maximization (EM) algorithm was used to detect the interpolation mode and filter coefficients. This method is vulnerable to tampering, since the edited image can be resampled to a target CFA. Similarly, in [15] an Support Vector Machine (SVM) was trained to predict the camera model used for acquisition. In [16], a known CFA pattern is used within an iterative process to impose constraints on the image pixels. These constraints are then used to check whether the image has undergone further manipulation.
Other works are devoted to a more realistic formulation of the problem. In [2], PRNU noise features and CFA interpolation coefficients are used jointly to estimate source type and camera model. In [17], an implicit grouping stage is added, under the assumption that each region is interpolated differently by the acquisition device depending on its structural features. The proposed system identifies 16 regions with an EM reverse classification algorithm and efficiently estimates interpolation weights. In [18], the concrete CFA configuration is determined (essentially the order of the sensed RGB components), in order to decrease the degrees of freedom in the estimation process.
Tampering is explicitly considered in [19], where a synthetic CFA is recreated to conceal traces of manipulation. Conversely, in [20] the presence of a realistic CFA is checked to distinguish real from PRCG images.

2.1.3 Lens characteristics

Each device model presents individual lens characteristics that can be used to link a particular device model to an image. In [21], lateral chromatic aberration is investigated. This lens aberration causes different light wavelengths to focus on shifted points of the sensor, effectively resulting in a misalignment between color channels. This is particularly apparent in low-end camera models, such as those embedded in mobile phones. The detected misalignment is fed into an SVM for classification.

In [22], radial distortion due to the lens shape is quantified using planar image regions. From the distortion parameters, clustering is performed since each camera has a characteristic distortion.
Lens characterization is pushed further in [23], where dust patterns are modeled by means of a Gaussian intensity loss model, resistant to watermarking and recompression, thus enabling the identification of a single device from an image.

2.1.4 Spatial-lighting transforms

Images are the end product of a physical acquisition process. Given an assumed reflection model, light color, position and intensity has to be consistent throughout the scene. Inconsistencies are indicative of tampering either from post-processing or as a result of photo recapture.
In [24], illuminant colors are estimated in inverse-chromaticity space. Inconsistencies between patches are found by estimating the distance in illuminant color between image patches. However, evaluation is empirical and not automatic. In [25], textured plane orientation is found by analyzing the nonlinearities introduced in the spectrum by perspective projection, which can be used to detect photo recapture. In [26], the first two orders of illumination spherical harmonics are extracted, according to a model approximating the illumination of Lambertian convex objects from distant sources. Inconsistencies are found in the image by comparing harmonic coefficients. Tampering is detected in [27] from specular highlights in the eye glints. The axis of illumination is found per glint to detect inconsistencies in the physical scene configuration.
In terms of individual camera footprints, each camera sensor has an individual radiometric response, which is normally shared across cameras of the same brand. This was characterized in [28] from a single greyscale image. It was also achieved in [29] with geometric invariants and planar region detection.
Finally, source classification is addressed in [30] where structural and color features are used to differentiate between real and computer generated images. PRCG recapturing attacks are examined and countermeasures provided.

2.1.5 D-A Reacquisition

One of the easiest methods to elude forensics analysis consists in recapturing forged and printed images. In these cases, the PRNU and CFA footprints of the camera would be authentic and all the low level digital detail would have been lost. Moreover, it is shown in [31] that people in general are poor at differentiating between originals and recaptured images, thus giving particular importance to photo recapture detection.
Some approaches have been devoted to recapture detection, which can be indicative of prior tampering. In [32], high frequency specular noise introduced when recapturing printouts is detected. A combination of color and resolution features are identified and used for SVM classification of original photos and their recaptured versions in [31]. In [33], a combination of specularity distribution, color histogram, contrast, gradient and blurriness is used.
The problem of original camera PRNU identification from printed pictures is studied in [34], highlighting the impact of unknown variables, including paper quality, paper feed mechanisms and print size.
Finally, a large database containing photo recapture from several widespread low-end camera models was presented in [35], and made publicly available for performance comparison.

2.1.6 Scanner acquisition

Similarly to camera footprints, scanner footprints can be used for device identification and linking. Moreover, scanned image tampering detection is of particular importance, since legal establishments such as banks accept scanned documents as proofs of address and identity [36].
In [37], noise patterns from different types of reference images are extracted in an attempt to extract a characteristic scanner PRNU equivalent. In [38], cases where scanner PRNU acquisition might be difficult are considered, e.g. due to the lack of uniform tones and the dominance of saturated pixels, such as in text documents. Image features based on the letter "e" are extracted, clustered together and classified with an SVM. Individual footprints are examined in [39], where scratches and dust spots on the scanning plane result in dark and bright spots in the image.
Source classification is also investigated. In [40], an SVM-based classification of PRCG, scanned and photographed images is made. Confusion between scanned and shot images was reduced due to physical sensor structure: cameras have a two dimensional sensor array, while scanners a one dimensional linear array, resulting in different noise correlation within the image. The same periodicity is exploited in [41], where camera-acquired and scanned images are classified with an SVM.

2.1.7 Rendered image identification

As PRCG images get more and more realistic, it becomes increasingly difficult to distinguish between real and synthetic images.
Different features have been employed to classify automatically PRCG and natural pictures. In [42], the main hypothesis is that statistical characteristics of residual noise is fundamentally different between cameras and CG software. Moreover, certain stochastic properties are shared across different camera brands, which cannot be found in CG images. This case does not cover the possibility of CG images recaptured with cameras. Based on the same approach, in [43] statistics of second order difference signals from HSV images are checked for classification. In [44], a combination of chromatic aberration and CFA presence in images is determined, as non-tampered PRCG images would not present CFA demosaicing traces. In [45], Hidden Markov Trees using DWT coefficients are employed to capture multi-scale features for PRCG/real image classification. Finally, in [30] a method is presented that takes into account a combination of features based on the inability of CG renderers to correctly model natural structures such as fractals and to reproduce a physically accurate light transport model, yielding classification accuracies of 83.5%.

2.2 Video Acquisition

Most, if not all, of the techniques developed for still images can also be directly applied to image sequences. As a consequence, the literature solely concerned with video acquisition is comparatively small.
One of the examples is the extraction of camera PRNU from video frames for video copy detection. In [46], the PRNU is extracted from video frames, in order to have an effective copy detection without getting the false positives due to videos shot from similar angles and different cameras. The estimated PRNU is averaged over the duration of a video and tested for robustness against blurring, AWGN addition, compression and contrast enhancement. In [47] and [48], the case of PRNU extraction from low resolution videos is considered, with emphasis on double compression with different codecs and YouTube uploading.
More specific to videos are the works presented in [49] and [50]. In the first paper, tampering is detected in interlaced and de-interlaced video through analysis of the fields. In interlaced videos,

motion across fields within the same frame and between neighboring frames should be identical. In deinterlaced videos, the correlations introduced by blending of the two fields can be corrupted by tampering. Also, an adaptation of the technique to reveal traces of frame rate conversion was presented. In the second paper, an SVM classifier was trained to recognize characteristic combing artifacts based on their neighborhood statistics.
A geometric approach is presented in [51], where reprojected video is identified from non-zero skew parameters being introduced within the camera intrinsic matrix. This process needs multiple frames from the same scene, which finds its ideal setting in the field of video forensic analysis.
Also specific to the video setting is the work presented in [52] and [53], due to the number of frames required by the proposed method. The method estimates a per-pixel noise function which is linear to the camera response function. Pixels that do not fit the linear correlation are automatically identified as forged.
Finally, in [54] the problem of pirating videos in cinemas is analyzed. The proposed method requires watermarked video projected in the cinema, and allows to recover the position of the pirate. A related paper [55] proposes a suitable watermark for the system, robust to geometric transformations and D-A and A-D conversion.

2.3 Audio Acquisition

Acquisition-based footprints in audio data constitute the most important cue for evaluating the authenticity of audio recordings. By recordings, we mean digitized acoustic signals (that may be speech, noises, music). In general, acquisition-based traces can not be found in synthesized audio data. In some cases, however, synthetic audio might be mixed from various sources, including previously recorded audio materials (commonly referred to as "audio samples"). In the following sections, we focus on surveying approaches for audio analysis that aim at characterizing the audio source and environment, the means of recording as well as the claimed recording time and place. Unique signatures are needed to authenticate digital audio data [56], [57]. More classic approaches dealing with analogue recording devices [58] will mostly be omitted here. These can be generated for example by microphone characteristics, the movement of recording and erase heads of analogue audio recorders or the electric network frequency.

2.3.1 Microphone classification

Analogous to image and video acquisition via camera devices, audio recordings of acoustic events can only be conducted via microphones. Thus, the literature gives some expamples of microphone identification. One recent approach is reported in [59], where the authors propose a context model for microphone recordings. It incorporates the involved signal processing chain and possible influence factors. Furthermore, a relatively extensive experiment is conducted to identify suitable classification schemes for pattern recognition of microphones. In total, 74 supervised classification techniques and 8 unsupervised clustering techniques are investigated. In these experiments, the second order derivative of Mel-Frequency Cepstral Coefficients (MFCC) [60] features exhibits the best discriminative power. The aforementioned work extends [61], where the authors follow a more basic approach and report promising results. As an acoustic feature, they extract histograms of FFT coefficients in time segments, where the digitized audio signal is almost silent, i.e. only the noise spectrum of the recording equipment is present. A variety of machine learning approaches is compared. With an empirically determined optimal noise threshold, the best classification results reach 93.5% accuracy when discriminating seven different microphones. The authors also report that PCA-based dimensionality reduction of the audio features is applicable without loss in accuracy.

Other previous publications of this group dealing with microphone and environment detection are [62] and [63]. Another work focussing on microphone identification is presented in [64]. The authors tested different classifiers and acoustic features to classify eight telephone handsets and eight microphones. Several cepstral features and derivatives thereof were evaluated. Conventional MFCCs resulted in the best trade-off between performance and feature dimensionality. The use of Gaussian supervectors as a statistical characterization of frequency domain information of a device contextualized by speech content was proposed. Thus, a template that captures the intrinsic characteristics of a device was obtained. Visualization of this template validated its discriminative power. A Support Vector Machine [65] classifier was used to perform closed-set identification experiments. The average identification accuracy for telephones was 93.2%. Interestingly the confusions were most common in the same transducer class (i.e., electrect vs. carbon-button). The average identification accuracy for microphones was reported with 99.0 %.

2.3.2 Electric Network Frequency

Electric network frequency (ENF) denotes the frequency of the AC power system, typically 50 or 60 Hz. The analysis of ENF has gained widespread use in the field of audio forensics research. On the one hand, traces of the electric network frequency are present in a multitude of audio recordings. On the other hand, the ENF exhibits characteristic fluctuations which are identical within a connected power grid. Consequently, the ENF information embedded in an audio recording may be used to determine the acquisition time of a recording. Publications covering the general use of ENF in forensic applications include [66, 67, 68, 69, 70, 71, 72, 73, 74, 56].

The frequency of the alternating current within connected power grids is held constant to a nominal value, for instance 50 Hz in Europe or 60 Hz in North America. However, due to changes in power generation and consumption, this frequency is subject to small alterations that occur as a function of time.

Typically, power grids covering large areas are operated in a synchronized, phase-locked fashion. Examples are the synchronous grid of Continental Europe, operated by the European Network of Transmission System Operators for Electricity (ENTSO-E), and the Eastern Interconnection and the Western Interconnection in North America. Due to this synchronization, the deviations of the network frequency are very stable throughout a connected grid. Experimental data comparing the trajectories of ENF at different places within synchronized power grids is given, for instance, in [66, 73, 75]. The magnitude of frequency deviations are relatively small, because the power grid operators control the power generation to hold this value within given bounds. According to the recommendations of the Union for the Co-ordination of Transmission of Electricity, the predecessor organization of ENTSO-E, alterations within $\Delta f \leq 50\,\text{mHz}$ fall into the normal operations range. While deviations $50\,\text{mHz} < \Delta f \leq 150\,\text{mHz}$ are considered acceptable, fluctuations above 150 mHz are not acceptable, since they pose severe risks of malfunctions in the electric power network (see [75]).

While some characteristic patterns are observable, for instance generated by periodic maintenance operations or network component switches [69, 75], the fluctuations of the electric network frequency are not predictable and appear as a random process. Thus, the variations of the electric network frequency measured over a significantly long time form a unique signature that can be used to determine the acquisition time of an audio signal.

ENF information is introduced into the audio signals in two principal ways. If the acquisition device is directly connected to the power grid, traces of its frequency are imposed on the recorded signal if non-ideal voltage controllers are used or due to magnetic interferences within the device.

In case of portable devices, the electromagnetic field of nearby supply lines or mains-powered de-

vices might superimpose on the audio recording. In [66, 71, 72], the radiation of different devices is investigated. For some devices, for instance incandescent bulb or fluorescent tubes, the ENF and its harmonics are clearly distinguishable in the spectrum. Other consumers, such as laptop computers, exhibit broadband electromagnetic fields that make an extraction of ENF components difficult. In [76], the sensitivity of acquisition devices to electromagnetic fields has been investigated in a controlled field. However, the results suggest that traces of ENF are detectable in the audio signal only if a dynamic, moving-coil microphone is used, while devices containing other types of microphone appear to be immune to such fields.

The generation of ENF components by a controlled magnetic field is also investigated in a study by Sanders and Popolo [74]. The influence of the magnetic flux density (measured in Gauss, unit Gs) is examined by exposing different recording devices in a controlled magnetic field generated by an electric coil. According to this experiment, a magnetic field with a flux density of 50 mGs does not cause any detectable ENF components compared to a measurement of the same device subject to only ambient magnetic fields. A flux density of 1 Gs leads to detectable traces of ENF for all devices. While such high flux densities may occur in close vicinity of electric devices such as power amplifiers, even the lower value of 50 mGs is unlikely in normal circumstances. As an example the authors state 1.0-6.5 mGs as typical values for office environments.

The extraction of ENF information from audio signals is based either on time-domain or frequency-domain methods. In the literature on ENF, often either three [69, 77] or four [75] different methods are mentioned. However, these are generally only small variations of the two general approaches or differ only in the analysis of the extracted data.

Frequency-domain methods for ENF are on based the short-time Fourier transform (STFT), which operates on (potentially overlapping) segments of the audio signal. To reduce the computational effort and the storage requirements, in particular if the obtained data is stored as a reference dataset, the signal is often downsampled prior to this operation, typically to sample rates around 300 Hz [72]. The length of the Fourier transform, the hop size determining the amount of overlap between subsequent Fourier transforms and choice of the window function are important parameters for the STFT operation that determines complexity, accuracy and the time resolution of the obtained ENF data. Since the ENF variations are very small, the time-frequency uncertainty principle (e.g. [78]) becomes a limiting factor in analyzing these data. To obtain a sufficient frequency resolution, very long FFT length are required, thus reducing the time resolution [77]. Alternative algorithms, namely the Chirp-Z transform and methods based on a eigendecomposition of a sample data covariance matrix, are proposed by the same author to improve the time or frequency resolution. In [72], a increase in frequency resolution is gained by zero-padding the audio segments and quadratic interpolation between FFT bins.

Time-domain methods measure the frequency by determining the period of an ENF oscillation. In [69, 75], this method is described as zero crossing detection. It offers high time resolution and high accuracy if the sampling frequency is sufficiently high. Band-pass filtering, in particular removal of DC components, and the use of interpolation techniques to determine zero crossing locations are crucial for high accuracy [69]. As a drawback, this method is limited to signals which contain only a single ENF component. Thus, it cannot be used if multiple traces of ENF, for instance from different modification or transmission steps, are contained in the signal. In [71, 76], the ENF is obtained in the time domain using a frequency counter, which is equivalent to counting the zero crossings.

Harmonics of the ENF fundamental frequency pose another way to determine the network frequency. Cooper [72] states that audio signals may contain harmonics with higher power than the fundamental frequency. At the same time, he considers it unlikely that any signal higher than the third harmonic can be used for analysis due to masking by the contained acoustic signal. Supporting this argument, [71] reports that the extraction of harmonics proved very difficult in the presence of speech signals.

In either case, no measurements about the relative power of the harmonics are provided. In [70]. The use of ENF harmonics and its relations to the the fundamental frequency to estimate properties of the recording equipment is suggested in [75].
To authenticate audio recordings, ENF variations have to be recorded and stored continuously for all synchronized power grids in question. Several attempts to create such databases are reported in the literature on ENF (e.g. [69, 72, 79, 73, 71]). However, it appears that still no coordinated archiving of ENF data takes place. Brixen [71] considers the use of data provided by power suppliers, but notes that these traces are typically stored for a limited time only.
To determine the acquisition time (and possibly place) of a signal, it must be compared to the ENF database. However, algorithms for matching ENF information gained relatively little attention so far. Often, the task of comparing and matching ENF plots is performed visually (e.g. [75, 80]). In [72], an automated approach based on a mean square error criterion is proposed. In [79], this mean square error approach is compared to a matching algorithm using autocorrelation coefficients. It is demonstrated that this approach yields significantly better results, especially if the tested audio segments are relatively short (e.g., below 10 minutes). In addition, the distance measure based on the autocorrelation is more robust to errors, for instance to static offsets of the ENF. Such errors may result from inaccurate sampling clocks in the acquisition device.

2.3.3 Environment Classification

The properties of the environment, namely the reverberation, form another part of the acquisition history embedded in an audio track. In the literature on audio forensics, the use of such footprints is handled only scarcely. In the context model for microphone forensics established in [59], the influenced of the room acoustic is modeled as an additional transfer function in the signal processing chain yielding a recorded microphone signal. In [63], the use of feature extraction and classification classification techniques to classify several features, including the reproduction room, is investigated. Due to the relatively low classification rates, the authors conclude that the influence of the recording room is often negligible compared to other influencing parameters such as the microphone used. However, this evaluation uses a black-box approach to evaluate different features and classification algorithms which does not take the particular characteristics of room acoustics into account. Thus, sensible approaches to detect environment footprints account for the characteristics of reverberation. In the analysis of gunshot recordings, e.g. [81, 82], reverberation is acknowledged to contain information about the environment and about the precise location of a shot. Nonetheless, it is usually regarded as clutter that hinders the retrieval of other information from these recordings.

Estimation of the Reverberation Time

One recent paper [83] considers the use of room acoustic parameters to authenticate digital recordings. In particular, the reverberation time is used as a parameter to characterize the recording room. While this approach appears to be unique within audio forensics research, measurement and estimation of the reverberation time are extensively investigated in general-purpose acoustics and acoustical signal processing. For the envisaged application, blind estimation methods are of particular interest, because they do not require dedicated measurements or particular test signals (within certain limits).
Two related approaches for the blind estimation of the reverberation time, which form the conceptual basis for [83], are [84, 85]. In these approaches, the reverberation of the recording room is modeled as a random process with exponential decay, which is uniquely determined by a time constant and an amplitude value. Thus, only the diffuse part of the reverberation tail is considered, while discrete

reflections are omitted. The time and amplitude parameters are estimated by a maximum likelihood estimator. It is reported that the quality of estimation depends on the input signals. Best results are obtained for sharp offsets in the source signal followed by periods of silence, which form periods of free decay in the recorded signal. On the other hand, segments of connected speech, speech onsets or gradually declining speech offsets degrade the accuracy of estimation. For this reason, additional processing of the obtained running estimate is necessary. This postprocessing step is implemented as an order-statistics filter [84] and consists of a histogram of previous estimates. The reverberation time corresponding to the first peak in this histogram is used as corrected estimate for the reverberation time parameter.
Different assumptions are made for the input signals. In [85, 83], the source signal is considered as a sequence of identical and independent normally-distributed random variables. In the same way, [84] assumes the reverberation tail to consist of uncorrelated noise with exponential decay and Gaussian distribution, although it is acknowledged that this model is highly simplified.

Estimation of the Room Impulse Response

Instead of characterizing the recording environment by single parameters such as the reverberation time, the room impulse response captures the complete acoustical transfer function of a room for given source and microphone positions. Thus, this impulse response can be used as an acoustic footprint to characterize the acquisition of an audio signal.
Blind dereverberation (e.g. [86, 87, 88]) is a field of active research which incorporates the estimation of the room impulse response. Dereverberation denotes approaches to remove components introduced by reverberation from an audio signals. Because the effect of reverberation can be modeled as a filtering process, dereverberation forms a special case of deconvolution [86, 89]. Corresponding algorithms generally estimate a model of the rooms impulse response, either in explicit form or implicitly in the adapted compensation filter. Blind dereverberation does not require a dry reference signal of the sound source. Typical applications of blind dereverberation include videoconferencing, automatic speech recognition or hands-free telephony [87, 90].
Auto-regressive (AR) models are the most common way to obtain a parametric model of the room impulse response, e.g. [86, 87]. Multichannel linear prediction is applied, amongst others, in [91, 89]. Blind dereverberation algorithms may also differ in the number of available microphones (or audio channels). The spatial diversity present in multiple input channels can be utilized to obtain more information about the source signal or the recording room [91, 87, 92]. In addition, algorithms are distinguished by the number of distinct sound sources present in the signal, e.g. [91, 88].
The spectral properties and the statistical model assumed for the sound source forms another distinction. In general, non-stationary (or time-varying) source characteristics are beneficial for a unique estimation of the room impulse response [86, 91]. Otherwise, the identification of the source and the room impulse response remains ambiguous.
Highly correlated source signals, for instance due to periodicity or harmonic contents, complicate application of conventional estimation techniques [90, 93]. In [91], a pre-whitening stage is proposed to eliminate correlations in the source signal, thus reducing the ambiguity between source characteristics and room transfer function. In [90], quasi-periodicity is introduced as an inherent property of speech signals. Two methods — one based on averaged transfer functions (ATF) and one based on a minimum mean squared error (MMSE) criterion — are proposed to account for the inherent periodicity of such signals. A recent approach [93] aims at dereverberation of music signals. It argues that the all-pole room transfer functions, which form the basis of AR models, are ill-suited for musical tones. Based on this deficiency, an algorithm based on Wiener filtering and Gaussian mixture modeling is proposed that accounts for the harmonic structure of musical contents. The algorithm

is tested on artificially reverberated monophonic MIDI signals as well as on tracks from commercial audio CDs. In both cases, the algorithm reduces the amount of reverberation significantly, and performs better than conventional algorithms based on inverse filtering.

Challenges and Advantages for Footprint Detection

Apparently, blind reverberation techniques are predominantly used with natural speech in real-time applications. Considering the application to audio tracks, which often contain musical contents and are typically generated by a sophisticated production process, this poses a number of new problems and challenges. First, the musical, often harmonic, nature of the signals limits the use of techniques exclusively targeting at speech signals. Second, musical recordings most often consist of many sound sources, which are typically recorded and processed separately. In addition, signal processing techniques such as equalization or artificial reverberation are often applied to these source signals before they are mixed into the final audio track. Thus, it may prove difficult to obtain a consistent room impulse response or reverberation time estimate from such content. The produced nature of audio tracks may also complicate the application of multichannel dereverberation techniques. In contrast to natural multi-microphone recordings, stereo or multichannel audio contents is typically generated by production techniques and may lack characteristics of natural multichannel recordings. On the other hand, the envisaged application offers a number of new possibilities. First, real-time capabilities are typically not required for footprint detection. Therefore, algorithms are not restricted by causality. Additionally, a higher computational effort is often permissible.

Chapter 3

Coding

3.1 Image Coding

Lossy image compression is one of the most common operation which is performed on digital images. This is due to the convenience of handling smaller amounts of data to store and/or transmit. Indeed, most digital cameras compress each picture directly after taking a shot. Due to its lossy nature, image coding leaves characteristic footprints, which can be detected.
JPEG is, by far, the most widely adopted image coding standard. Section 3.1.1 briefly summarizes the main processing steps performed by JPEG compression and describes methods that can be adopted to discriminate JPEG-compressed images from uncompressed images. When JPEG compression is detected, we also discuss methods that estimate the coding parameters used at the encoder. Due to the ease in manipulating digital content, images might go through one or more compression steps. In Section 3.1.2 we describe methods that are able to detect whether an image has been compressed once or twice. Different cues related to double JPEG compression have been exploited in the past literature, ranging from the structure of histograms of quantized DCT coefficients, to image statistics and blocking artifacts. Many image coding schemes, including JPEG, operate on images in a block-wise fashion. As such, blocking artifacts appear in the case of aggressive compression. Section 3.1.3 illustrates methods aimed at detecting blockiness in lossy compressed images. In order to contrast the applicability of the aforementioned methods, a knowledgeable adversary might conceal the traces of coding-based footprints. Section 3.1.5 summarizes the anti-forensic techniques that have been recently proposed for this purpose. Although revealing coding-based footprints in digital images is in itself relevant, coding-based footprints are fundamentally a powerful tool for detecting forgeries [94][5]. We refer the reader to Chapter 4 for a detailed description of forgery-detection methods, including those that leverage coding-based footprints.

3.1.1 JPEG

Nowadays, JPEG is the most common and widespread compression standard [95]. The standard, originally proposed by the Joint Photographic Experts Committee, specifies two compression schemes, lossy and lossless, although the former is, by far, the most widely adopted. According to the specifications of the lossy scheme, JPEG converts color images into a suitable colorspace (e.g. YC_bC_r), and processes each color component independently (after spatial subsampling of the chroma components). Without loss of generality, in the following we refer to the compression of the luma component, unless stated otherwise. Compression is performed following three basic steps:

- Discrete Cosine Transform (DCT): an image is divided into 8×8 non-overlapping blocks. Each block is shifted from unsigned integers with range $[0, 2^b - 1]$ to signed integers with range $[-2^{b-1}, 2^{b-1} - 1]$, where b is the number of bits per pixel (typically $b = 8$). Each block is then DCT transformed in order to obtain the coefficients $Y(i, j)$, where i and j are the row and column indexes for the elements within a block.

- Quantization: the DCT coefficients obtained in the previous step are quantized according to a quantization table which must be specified as an input to the encoder. Quantization is defined as division of each DCT coefficient $Y(i, j)$ by the corresponding quantizer step size $\Delta(i, j)$, followed by rounding to the nearest integer. That is,

$$Z(i, j) = \text{sign}(Y(i, j))\text{round}\left(\frac{|Y(i, j)|}{\Delta(i, j)}\right). \tag{3.1}$$

  Thus, the reconstructed value at the decoder is

$$Y_{\mathcal{Q}}(i, j) = \Delta(i, j) \cdot Z(i, j). \tag{3.2}$$

  The quantization table is *not* specified by the standard. In many JPEG implementations, it is customary to define a set of tables that can be selected specifying a scalar quality factor Q. This is the case, for instance, of the quantization tables adopted by the Independent JPEG Group, which are obtained by properly scaling the image-independent quantization table suggested in Annex K of the JPEG standard with a quality factor $Q \in [1, 100]$.

  The purpose of quantization is to achieve compression by representing DCT coefficients at a target precision, so as to achieve the desired image quality. Since quantization is not invertible, this operation is the main source of information loss.

- Entropy Coding: DCT quantized coefficient are lossless coded and written to a bitstream. A common coding procedure is variable length coding by means of properly designed Huffman tables.

In many situations, one has access only to the decoded image (i.e., the image is available in the pixel domain only). In these cases, it is unknown whether the image had been previously compressed and, in this case, which were the compression parameters being used. This might be a useful information, e.g., when processing the image to eliminate blocking artifacts, or determining the distortion with respect to the original image in a no-reference fashion. In the following we discuss several methods that have been proposed to identify the compression history.

Algorithms for the identification of compression history

When an image is available in the pixel-domain, a challenging problem consists in studying its compression history. That is, exploiting the knowledge of raw pixel values, it is possible to detect whether an image has been previously compressed. The key tenet is that block-based image coding leaves characteristic compression artifacts that can be revealed.
To this end, the authors of [96] and [97] describe a method capable of revealing artifacts also when very "light" JPEG compression is applied, e.g. when using a quality factor Q as high as 95. This algorithm exploits a very simple idea: if the image has not been compressed, the pixel differences across blocks should be similar to those within blocks. Then, it is possible to build two functions, Z' and Z'', taking into account inter and intra-block pixel differences. The difference between

the histograms of Z' and Z'' is compared to a threshold in order to reveal the presence of prior compression.
In [98] the authors observe that in a JPEG-compressed image, the integral of the DCT coefficient histogram in the range $(-1, +1)$ is greater than the integral in the range $(-2, -1] \cup [+1, +2)$. If the ratio between the first and second integral is close to 0, the image has probably been JPEG-compressed. Hence, they propose to set a threshold on the value of this ratio, so that JPEG compression is detected when the ratio is smaller than the threshold. Additionally, they propose algorithms for estimating the JPEG quantization steps of a compressed image, and detecting the quantization table of a JPEG image. It is also interesting to note that the performed analysis is based on both the quantization and rounding errors.
A different, more general, approach is discussed in [99], where the aim is to estimate the class of image coding schemes being used, together with the adopted parameters. In this respect, three different coding schemes are considered: transform coding (DWT, DCT), subband coding and differential image coding (DPCM). The algorithm is composed of many steps. The first one consists in finding the presence of footprints left by a general block-based encoder. To this end, the gradient between adjacent pixel values is computed. The periodicity of the gradient is an evidence of a block-based editing, and the period is directly proportional to the block size. For this reason, the larger the size of the block, the smaller the number of periods, and consequently the harder is to estimate the block size. If evidence of block-based coding is found, a similarity measure for each coding scheme is computed in order to detect the one being used. Similarity measures are based on footprints left by each coding method. As a matter of fact, transform coding is characterized by comb-shaped histograms of the coefficients in the transform domain. Subband coding is characterized by the presence of ringing artifacts near image edges. Differential image coding is characterized by the whiteness of the residual obtained from the difference between the encoded image and its denoised version. Special attention is paid to H.264 intra prediction. The method giving the highest similarity measure is selected.

Algorithms for the estimation of quantization step

After identifying an image as JPEG-compressed, an interesting problem consists in finding the parameters used during compression. In the case of JPEG, this means estimating the used quality factor Q or the whole quantization matrix $\Delta(i, j)$, $1 \leq i, j, \leq 8$. Most of the methods proposed in the literature observe the fact that the histogram of DCT coefficients has a characteristic comb-like shape, where the spacing between successive peaks is related to the adopted quantization step size. The work in [96] and [97] proposes to exploit a distinctive property of the histogram of DCT coefficients. Specifically, it shows that the envelopes of such histograms can be approximated by means of a Gaussian distribution for DC coefficients, and a Laplacian distribution for AC coefficients. Leveraging this observation, the quality factor is estimated through a maximum likelihood (ML) approach. In order to estimate the parameters of the probability distribution, uniform and saturated blocks are excluded from computations. These blocks are easily identified by checking the minimum and maximum pixel values. If they coincide, the block is uniform; if they reach the actual minimum and maximum values (0 and 255 for an 8-bit image), the block may contain truncated pixels.
In [100] the authors propose a method for estimating the elements of the whole quantization table. To this end, separate histograms are computed for each DCT coefficient subband. Analyzing the periodicity of the power spectrum, it is possible to extract the quantization step $\Delta(i, j)$ for each subband. Periodicity is detected with a method based on the second order derivative applied to the histograms.
In [99], another method based on the histograms of DCT coefficients is proposed. There, the

authors estimate the quantization table as a linear combination of existing quantization tables. A first estimate of the quantization step size for each DCT band, i.e. $\Delta(i,j)$, is obtained from the distances between adjacent peaks of the transformed coefficients histograms. However, in most cases, high-frequency coefficients do not contain enough information. For this reason some elements of the quantization matrix cannot be reconstructed, and they are estimated as a linear combination (preserving the already obtained quantization steps) of other existing quantization tables collected into a database.
When a JPEG image is reconstructed in the pixel domain, pixel values are rounded to integers. As a consequence, the histograms of DCT coefficients ($\hat{Y}_{\mathcal{Q}}(i,j)$) computed from decoded pixel values are not exactly comb-shaped, in the sense that they are blurred with respect to those obtained directly after quantization ($Y_Q(i,j)$). Indeed, due to rounding, $\hat{Y}_{\mathcal{Q}}(i,j) \neq Y_{\mathcal{Q}}(i,j)$. In [98] the authors address this issue, by showing that rounding the reconstructed DCT coefficients $\hat{Y}_{\mathcal{Q}}(i,j)$ reduces significantly the spreading effect on the peaks. In this way it is possible to estimate the quantization step for each DCT frequency by looking at peaks distances in such rounded-coefficients histograms.
In the case of color image compression, distinct quantization tables can be used for each color component. In [101] the authors target the problem of estimating these quantization tables. First, they introduce a MAP estimation method for extracting the quantization step size in grayscale images, exploiting the periodicity of DCT coefficients histograms. This is a refinement of the algorithm already discussed in [97]. Then, they extend the solution to color images. In this situation, the periodicity of the histogram is revealed only when the image is transformed to the correct colorspace and interpolation artifacts are removed. To this end, they propose a dictionary-based method to find both the correct colorspace and the interpolation method. After this pre-processing step, the MAP estimation method can be applied at each color component independently.

3.1.2 Double JPEG

Among the strategies addressing the estimation of coding parameters for compressed images, it is worth considering those solutions which are able to detect multiple compressions of the same image. These algorithms aim at detecting whether a compressed image has been coded twice and at identifying the coding parameters of the first step. Indeed, the parameters related to the second step can be obtained directly from the coded bitstream. The application scenarios of these strategies are diverse, including, e.g.: i) steganalysis [102], since steganographic programs perform a second compression after the hidden image is fused in the original one; ii) detection of image manipulation [103], when the original image is decompressed, modified, and recompressed; iii) forgery identification [104]; and iv) quality assessment [103].
According to the coding scheme summarized in the previous section, JPEG compression proceeds quantizing the transform coefficients resulting from block-wise DCT. As a matter of fact, double JPEG compression can be approximated by double quantization of transform coefficients $Y(i,j)$, such that:

$$Y_{\mathcal{Q}_1,\mathcal{Q}_2}(i,j) = \Delta_2(i,j) \cdot \text{sign}(Y(i,j)) \text{round}\left(\frac{\Delta_1(i,j)}{\Delta_2(i,j)} \text{round}\left(\frac{|Y(i,j)|}{\Delta_1(i,j)}\right)\right), \tag{3.3}$$

with

$$Y_{\mathcal{Q}_1}(i,j) = \Delta_1(i,j) \cdot \text{sign}(Y(i,j)) \text{round}\left(\frac{|Y(i,j)|}{\Delta_1(i,j)}\right), \tag{3.4}$$

where the coefficients $Y_{\mathcal{Q}_1}(i,j)$ are obtained from the first compression using the quantization step Δ_1, and coefficients $Y_{\mathcal{Q}_1,\mathcal{Q}_2}(i,j)$ are obtained from the second compression using quantization step Δ_2.

Experimental results have shown that double quantization significantly affects the statistics of the transform coefficients. Moreover, pixels in the reconstructed image present some characteristic features that are altered whenever a second coding step is performed. Assuming that the original transform coefficients are characterized by a Laplacian distribution, it is possible to identify distinctive patterns in the resulting statistics that enable the identification of double compression, together with the values of the quantization steps that were adopted. To this purpose, the approaches proposed in the literature follow different strategies, which are summarized in the following.

Algorithms based on coefficient histograms

A major set of solutions include all those algorithms that rely on the shape of the histogram of DCT coefficients. In [103], Lukáš and Fridrich show how double compression introduces peaks in the histogram, which alter the original statistics and assume different configurations according to the relationship between Δ_1 and Δ_2. More precisely, the authors highlight how peaks can be more or less evident depending on the relation between the two step sizes, and propose a strategy to identify double compression. Specifically, special attention is paid to the presence of the so-called double peaks and missing centroids (those with very small probability) in the DCT coefficient histograms, as they are said to be robust features providing information about the primary quantization. Their approach relies on cropping the reconstructed image (in order to disrupt the structure of JPEG blocks) and compressing it with a set of candidate quantization tables. The image is then compressed using Δ_2 and the histogram of DCT coefficients is computed. The estimator chooses the quantization table such that the resulting histogram is as close as possible to that obtained from the reconstructed image. An implementation of this method is described in [105], providing a way to automatically detect and locate regions that have gone through a second JPEG compression. Note that the method is similar to JPEG calibration [106], a technique adopted in image steganalysis in order to obtain an estimate of the cover image.
A similar solution is proposed in [102], which considers only the histograms related to the 9 most significant DCT subbands, which are not quantized to zero. The corresponding quantization steps (employed in the first quantization) are computed via a Support Vector Machine (SVM) classifier. The remaining quantization steps are computed via a Maximum Likelihood estimator.

Algorithms based on idempotency of coding operators

A second set of strategies rely on the property of idempotency that characterizes the operators involved in the coding process. More precisely, these solutions assume that re-applying the same coding operations on the reconstructed test image would lead to a new image that results to be highly correlated with the image under examination. As a matter of fact, it is possible to identify the correct operator (or the codec) by evaluating which coding operation maximizes the correlation between the analyzed and reconstructed image after re-compression. In [107] Xian-zhe *et al.* present a method for identifying tampering and recompression in a JPEG image based on the requantization of transform coefficients. The main idea relies on the fact that, in case the image has been compressed twice and the analyst identifies the right quantization steps of the first compression, most parts of the reconstructed image result to be highly-correlated with the analyzed image. However, some parts of the image might exhibit poor correlation. These parts are identified comparing the value of a similarity measure, e.g., the Sum of Absolute Differences (SAD), with a given threshold and then labeling them as tampered regions.

Algorithms based on natural statistics of images

Another class of detection algorithms for double compressed images are based on checking whether an image presents some statistical properties that are expected to be found in natural imagery after one compression step. In [108], Fu *et al.* present a detection approach based on the analysis of the histogram of the first digit of the unsigned decimal representation of quantized DCT coefficients. They observe that the histogram obeys the generalized Benford's law [109]. That is,

$$p(d) \simeq \log_{10}\left(1 + \frac{1}{s + d^{\alpha}}\right) \qquad d = 1, \ldots, 9, \tag{3.5}$$

where s, α are parameter values that determine the model, d is the value of the first digit and $p(d)$ is the normalized histogram. Experimental results have shown that each compression step alters the statistics of the first digit distribution. As a consequence, the fitting provided by the generalized Benford's equation (3.5) is decreasingly accurate with the number of compression steps.
Starting from this experimental evidence, Li *et al.* [110] employ this property to identify double JPEG-compressed images. More precisely, the proposed approach computes the histogram of the first digit extracted from quantized coefficients. Then, the model in (3.5) is fitted to the actual data and a divergence measure is computed between the experimental normalized histogram and the model. By computing separate models and divergence values at the different DCT subbands, the authors are able to design an effective detection strategy. The estimation is performed considering only the DC coefficient and the low frequency AC coefficients, since the coefficient statistics for the remaining subbands prove to be highly varying and less reliable.

Algorithms based on analysis of coding artifacts

Another class of strategies include all those solutions that detect double compression by identifying coding artifacts that do not suit the coding operations of the second compression. As an example, it is possible to exploit blocking artifacts in order to understand whether the reconstructed image has been compressed twice. These solutions rely on the fact that blocks might not be aligned in two successive coding operations. Therefore, blocking artifacts are spread over the whole image instead of being localized along block borders.
The approach proposed in [111] computes a blocking artifacts map on the reconstructed image and feeds each region in the picture to a neural network that is able to identify whether the region has been recompressed and/or tampered. This method requires the pasted region to be extracted from a JPEG compressed image with quality factor smaller than that of the resulting tampered image, and it is based on the assumption that the grids of the original image and the source image (i.e., the image the pasted region is extracted from) are mis-aligned. Region partitioning is introduced to reduce the computational complexity and improve performance, and it is performed via segmentation of the tampered image.

Algorithms analyzing alterations in the spatial distribution of coefficients

Other solutions start from the assumption that modifying a compressed image and recompressing it leads to the alteration of the spatial distribution of coefficients. As a matter of fact, it is possible to discover alterations by assuming that the image signal is the result of the superposition of different components that are mixed together in the resulting image. By means of Independent Component Analysis (ICA), it is possible to identify the different contributions and separate them into independent signals. The solution proposed by Qu *et al.* [112] is based on a convolutive mixing model that identifies which part of an image presents coefficients that have been cropped in the spatial domain.

This assumption is motivated by the fact that blocks are normally misaligned when copying parts of an image into another one, and therefore, transform coefficients in the tampered area present a different spatial distribution. This effect can be modeled as the convolution of two independent signals that can be separated performing ICA.
Similarly, the approach [113] exploits the alterations in the periodicity of the signal brought by tampering with the image in two different domains, i.e. both in the spatial and in the transform (DCT) domains. Specifically, the periodicity in the spatial domain is studied based on the analysis of the blocking artifacts, while the periodicity in the DCT domain is due to the DCT coefficients being multiple of the quantization step. In both cases the recompression changes the periodic characteristics of the considered image. One interesting point of the proposed techniques is that they are able to work with both block-aligned and misaligned recompression. The proposed methods are used for detecting cropping and recompression attacks, and for detecting composite JPEG sources.

3.1.3 Blockiness

JPEG compression belongs to the broader family of block-based image coding schemes. When images are compressed at low rates, blockiness is introduced, in the form of visible discontinuities at block boundaries. Then, it is possible to exploit this footprint in order to understand whether an image has been block-processed. Indeed, several methods aiming at estimating blockiness are proposed in the literature.
In [114] the authors model a blocky image as a non-blocky image interfered with a pure blocky signal. Then, the estimation of blockiness in a blind way is turned into the problem of estimating the presence of the blocky signal without accessing the original image. In order to achieve this goal, the absolute value of the gradient between each column or row of the image is computed separately. Each one of these 2D signals is rearranged into a 1D signal, and its power spectrum is computed by means of the FFT. It is shown that the power spectrum of a blocky image presents some periodic spikes which are not present in the power spectrum of the non-blocky image. By knowing this, the power of the blocky signal can be estimated in order to reveal its presence. The proposed model allows the integration of features that reflect the human visual perception of the blocking artifacts.
In a similar way, a method for no-reference blockiness estimation is presented in [115]. First, block size and block locations are identified, accommodating the case of rectangular blocks (number of rows different from the number of columns). In this respect, the vertical and horizontal gradients are computed and their periodicity due to gradient peaks at block boundaries is also estimated in the frequency domain using the DFT. Such periodicities are directly proportional to the vertical and horizontal block size, while gradient peak locations enable estimating block positions. After the block localization step, a metric for blockiness distortion evaluation is computed, employing a weighting scheme based on a simple model of the Human Vision System. In particular, the authors use the local gradient energy as a metric, normalized by its neighboring pixels, that is computed separately on each dimension.
Tjoa *et. al* propose in [116] another method exploiting the periodicity of the directional gradient to estimate the size of the processed blocks. In particular, the authors subtract a median filtered version to the gradient, in order to enhance the peaks, and then apply a threshold based on the sum of the gradients, aimed at avoiding spurious peaks caused by edges from objects in the image. The period of the resulting function is computed using a maximum likelihood estimation scheme commonly adopted for pitch-detection. Moreover, a binary hypothesis test that measures the accuracy of the estimate is described.
A different approach is developed in [117], where as a first step a measure of the blockiness of each pixel is calculated applying a first order derivative in the 2D spatial domain. From the absolute value

of this measure, a linear dependency model of pixel differences is carried out for the within-block and across-block pixels. In order to estimate the probability of each pixel following this model, an EM algorithm is used. Finally, by computing the spectrum of the probability map obtained in the previous step, the authors extract several statistical features, and estimate the blocking periodicity of the blocky image.

3.1.4 Wavelet-based coding

JPEG is the most widely adopted image coding standard. However, during the last two decades, several other image coding architectures have been proposed with the goal of improving the coding efficiency and/or addressing additional requirements (e.g. spatial scalability, quality scalability). In this context, wavelet-based coding schemes offer an important alternative, which has led to the definition of the JPEG2000 standard. Unlike JPEG, wavelet-based codecs apply the transform to the whole image, or to large image tiles, in order to avoid blocking artifacts. Similarly to JPEG, lossy compression in achieved by quantizing transform coefficients. To the authors' knowledge, [99] is the only work that proposes a method to distinguish between DCT and DWT based coding. As discussed above, the method relies on detecting blocking artifacts introduced by DCT-based coding schemes.
Following the same approach as [103], the work in [118] analyses the differences in the DWT sub-bands between single and double JPEG2000 compression. A different approach is pursued in [119]: prediction residuals obtained subtracting horizontal, vertical and diagonal neighbors are modeled by means of a Markov random process in order to capture the inter-pixel correlations statistics, which are fed to a SVM classifier to detect double JPEG2000 compression.

3.1.5 Anti-forensics

In the previous sections we described several methods that can be used to trace back the processing history of images, exploiting coding-based footprints. Such footprints can also be leveraged by the forensic analyst to identify traces of forgeries, e.g. local tampering, copy/move forgeries, etc. Therefore, a knowledgeable adversary might want to conceal these footprints in order to fool the methods described above. Recently, a lot of attention has been focused on anti-forensic methods of this sort.
Stamm *et. al* propose in [120] a method for removing the quantization artifacts left on DCT coefficients in JPEG-compressed images. The main idea is to modify the comb-shaped distribution of DCT coefficients in JPEG-compressed images, in such a way to restore a Laplacian distribution, which typically arises in uncompressed natural imagery. The proposed anti-forensic method adds a dithering noise signal in the DCT domain. To this end, the model parameter of the underlying Laplacian distribution is estimated for each DCT subband (excepting the DC component, where no parameters are estimated). Then, a suitable dithering noise distribution is designed (for the DC component a uniform distribution is used over each quantization interval), and the corresponding noise samples are added to the DCT coefficients.
The same authors propose a deblocking method to remove blocking artifacts caused by JPEG compression in [121]. The deblocking operation consists in smoothing the JPEG-compressed image with a median filter, and then adding a low-power white noise signal across the image. By properly choosing the correct parameters for the filter and the noise, it is possible to mislead the JPEG compression identification method proposed in [97]. Similarly, by combining the above anti-forensic technique [120] and this deblocking operation, the digital image forgery detector proposed in [100] will fail.

In [122], the very same method is extended and generalized to quantization footprints left by a wavelet-based coding scheme (e.g. JPEG2000). Similarly to the DCT case, the histograms of discrete wavelet transform (DWT) coefficients appear to be comb-shaped, due to quantization. This feature was, for example, used in [99] to reveal the presence of DWT compression. However, it is also shown that by adding a dithering noise signal sampled from a specific probability density function to the DWT coefficients, it is possible to obtain a histogram that can be confused for that of an uncompressed image.

3.2 Video Coding

3.2.1 Parameter estimation

Departing from initial works focused on detecting coding footprints on static images, new algorithms have been developed, specifically targeting coded video content. Among these it is possible to include the recent works in the field of multimedia forensics and no-reference quality assessment, which aim at estimating the coding parameters and the characteristics of the coded video data. The choice of coding parameters in relation to the characteristics of the video content is ruled by non-normative coding tools which can vary according to the specific implementation of the video codec. This piece of information might enable the identification of the vendor-dependent implementation of the codec, which can be used, for example, to verify intellectual property infringements, to identify the codec used to produce the content, and to estimate the quality of the reconstructed video without the availability of the original source.
In previous works, estimation of encoding parameters has been mainly employed to address the problem of no-reference quality assessment. It is possible to cluster the proposed solutions in different classes: i) methods that perform parameter estimation by re-encoding the reconstructed content; ii) methods that analyze the DCT coefficients of the reconstructed content, focusing on the comb-like shape of the histograms, in order to identify which quantization step size was used; iii) methods that analyze coding artifacts, trying to reveal which coding operations have generated them; iv) methods that infer the characteristics of the original content from the coded syntax elements. The last class includes approaches that are significantly different from those of the other classes, which do not assume the availability of the coding parameters (e.g. QPs or motion vectors). However, it is worth mentioning them for the sake of comparison.

Solutions based on re-coding

The work described in Chen *et al.* in [123] targets the estimation of the coding parameters for MPEG-2 coding. The proposed approach relies on the assumption that, in case the reconstructed sequence is re-compressed, the encoder is likely to choose the same configuration for most of the coding parameters. This natural tendency is motivated by the characteristics of the coded signal and by the coding footprints left in the reconstructed sequence by the first coding step. The proposed approach processes the video sequence as follows. At first, it detects whether the sequence has been compressed or not. Then, it identifies intra-coded frames by analyzing the characteristics of coefficient histograms (similarly to the JPEG case). Finally, it estimates quantization parameters.

Methods based on analyzing the statistics of DCT coefficients

As shown in the Section 3.1, quantization operates a decimation in the set of values that transform coefficients might assume. This results in a comb-shaped histogram of values where the distance

between non-zero bins permits inferring the value of the adopted quantization step. This property is exploited both in [123] and in [124]. It also proves to be robust to possible anti-forensic attacks performed by a knowledgeable adversary who aims at hiding the traces of compression [120]. The comb-shaped characteristics of coefficient histograms permits estimating motion vectors as well. Indeed, in [125] Valenzise *et al.* are able to estimate motion vectors by searching the predictor block whose related residual block presents this distinctive feature in the histogram of DCT coefficients.

Methods exploiting artifacts

Other solutions rely on detecting coding artifacts (like blockiness) and infer the coding parameters from them. As an example, it is possible to consider the approach in [126] where block sizes are estimated by analyzing the reconstructed picture in the frequency domain and detecting those peaks that alter the natural shape of the signal, since they are related to the presence of a block boundaries (enhanced by quantization). The quantization step size is computed by identifying the value that maximizes the similarity of the associated coefficient with the reconstructed one.

Methods recovering the statistics of the original signal

A fourth class of solutions aim at estimating the statistics of the original signal from the coded data. In this case, the syntax elements of the coded bit stream are available (quantization steps, motion vectors, quantized coefficients) and the main purpose is to recover the statistics of the compressed data steam. These solutions are mainly employed for no-reference PSNR estimation, since the original signal is not available. As such, they are different from the methods discussed above, in which coding parameters are not known. A first approach was proposed by Ichigaya *et al.* [127], where the moments computed based on the distribution of compressed coefficients are mapped to the statistics of the original transform coefficients. The obtained statistics are compared with that of the reconstructed coefficients to compute the source coding quality. A second approach was presented in [128], where a MAP estimation of the coefficient statistics is combined with a perceptual model in order to obtain an effective quality metrics that does not require the original reference. Most of these methods assume that coefficients are distributed according to an *a priori* model based on the Laplacian distribution, but other distributions are considered as well (e.g. Cauchy, Gaussian, generalized Gaussian, etc.).

3.2.2 Network footprints

Among the characterizing footprints which are left in reconstructed video content, some can be related to the transmission over a noisy channel. Indeed, packet losses and errors might affect the received packet stream. As a consequence, some of the coded data will be missing or corrupted. Error concealment is designed to take care of this, trying to recover the correct information and mitigate the channel-induced distortion. However, this operation introduces some artifacts in the reconstructed video which can be traced back and permits inferring the underlying loss (or error) pattern. The specific loss pattern permits identifying the characteristics of the channel that was employed for the transmission of the coded video. More precisely, it is possible to analyze the loss (error) probability, the burstiness, and other statistics related to the distribution of errors in order to identify, e.g., the transmission protocol or the streaming infrastructure. Most of the approaches targeting the identification of network footprints are intended for the estimation of the quality of the final video sequence (without having the original source as reference). These solutions are designed to provide network devices and client terminals with effective tools that measure the Quality-of-Experience (QoE) offered to the end user.

The proposed approaches can be divided into two main groups. The first include those solutions that try to infer network footprints using loss (error) pattern and network statistics in addition to the reconstructed data. The second set comprises strategies that process the reconstructed video sequences in the pixel-domain.

Approaches based on transmission statistics

The first class of network footprint identification algorithms takes into consideration transmission statistics to estimate the channel distortion produced on the reconstructed sequence. In [129] Reibman and Poole present an algorithm based on several quality assessment metrics (like SSIM, MSE, and SBM) to estimate the packet loss impairment in the reconstructed video. However, the proposed solution adopts full-reference quality metrics that require the availability of the original uncompressed video stream. A different approach is presented in [130], where the channel distortion affecting the received video sequence is computed according to three different strategies. A first solution computes the final video quality from the network statistics; a second solution employs the packet loss statistics and evaluates the spatial and temporal impact of losses on the final sequence; the third one evaluates the effects of error propagation on the sequence. These solutions are targeting control systems employed by network service providers, which need to monitor the quality of the final video sequences without accessing the original signal. Similarly, Naccari *et al.* [131] present another no-reference PSNR estimation strategy that does not require to compare the reconstructed sequence with the original one. The proposed solution evaluates the effects of temporal and spatial error concealment and the output values present a good correlation with MOS scores. As a matter of fact, it is possible to consider this approach as an hybrid solution, in that it exploits both the received bitstream and the reconstructed pixel values.

Approaches based on the reconstructed signal only

A second class of strategies assume that the transmitted video sequence has been decoded and that only the reconstructed pixels are available. This situation is representative of all those cases where the video analyst does not have access to internal parameters and information of set-top boxes and transmission devices. The solution proposed in [132] builds on top of the metrics proposed in [131], but departs from it since no-reference quality estimation is carried on without considering availability of the bitstream. Therefore, the proposed solution processes only the pixel values, identifying which macroblocks were lost, and producing as output a quality value that presents a good correlation with the MSE value obtained in full reference fashion.

3.2.3 Double compression

Every time a video sequence that has already been compressed is modified, it has to be re-compressed. Studying processing chains consisting of multiple compression steps is useful, e.g., for tampering detection or to recognize the original encoder being used. This is the typical situation that we face when video content is downloaded from video-sharing websites. Of course, it is possible to obtain the parameters used in the last compression step, as they can be read directly from the bitstream. However, it is much more challenging to extract information about the previous coding steps. For this reason some authors have studied the footprints left by double video compression. The solutions proposed so far in the literature are mainly focused on MPEG video, thus exploiting some of the properties already discussed in the case of JPEG double-compression.
In [104] the authors address the problem of estimating the traces of double compression of an MPEG coded video. The problem is faced either when the Group of Pictures (GOP) structure is preserved

or not. In the former situation, every frame is re-encoded in a frame of the same kind, so that I,B, or P frames remain, respectively, I,B, or P. The latter situation is typical in frame removal or insertion attacks. Since encoding I-frames is not dissimilar from JPEG compression, when an I-frame is re-encoded at a different bitrate, DCT coefficients are subject to two levels of quantization. Therefore, similarly to JPEG double compressed images, the histograms of DCT coefficients assume a characteristic shape that deviates from the original, e.g. Laplacian, distribution. In particular, when the quantization step size decreases from the first to the second compression, some bins in the histogram are left empty. Conversely, when the step size increases, some spikes appear in the histograms. When the GOP structure is changed, I-frames can be re-encoded into another kind of frame. However this gives rise to larger prediction residuals after motion-compensation. The authors show that by looking at the Fourier transform of the energy of the displaced frame difference over time, the presence of spikes reveals a change in the GOP structure, which is a cue of double-compression.
In [133] the authors propose another method for detecting MPEG double compression based on blocking artifacts. A metric for computing the Block Artifact Strength (BAS) for each frame is defined. This score is inspired by the method in [97] and relies on the difference of pixel values across a grid. Given a sequence, the mean BAS is computed for sequences obtained removing from one to eleven frames, obtaining the so called *feature vector*. If the sequence has been previously tampered with by frame removal and re-compression, the *feature vector* presents a characteristic behavior.
In [134], MPEG double quantization detection is addressed on a macroblock-by-macroblock basis. In particular, a probability distribution model for DCT coefficients of an I-frame macroblock is discussed. With an Estimation-Maximization (EM) technique, the probability distribution that would arise if a macroblock were double-quantized is estimated. Then, such distribution is compared with the actual distribution of the coefficients. From this comparison, the authors extract the probability that a block has been double-compressed.

3.3 Audio Coding

Similarly to images and video, audio files are typically compressed using lossy coding techniques. Section 3.3.1 provides a short overview of the lossy audio coders. Algorithms for the identification of the type and the parameters of the audio coder are discussed in Section 3.3.2.

3.3.1 Lossy audio coders

Lossy audio compression is used in a wide range of applications. Two of the most common implementations can be found in the MPEG AAC and in the MPEG-1/2 Layer 3 (MP3) [135] audio coders. Both of them adhere to the classic scheme of perceptual audio coding. A basic perceptual audio coder consists of four fundamental building blocks: the analysis filter bank, the psychoacoustic (i.e., perceptual) model, the quantization and coding block and the bit stream formatting [136], which is shown in Figure 3.1.
Since the quantization is preceded by subband filtering or a transform of the signal from the time into the time/frequency domain using a so-called Modified Discrete Cosine Transform (MDCT) [137] filter bank, these coders are also referred to as transform or subband coders. The transforms or subband decompositions also reduce the correlation in the signal, because of the downsampling after the subband filtering in the filter bank. Therefore, they are also used to reduce redundancy in the signal, based on the so-called transform coding gain. The basic principle of transform coding is

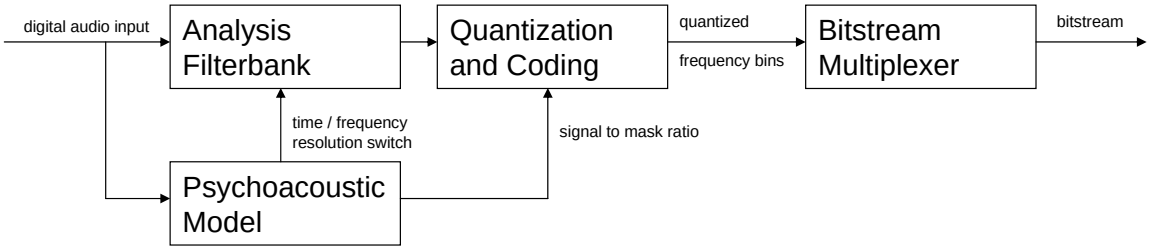

Figure 3.1: Basic perceptual audio coder

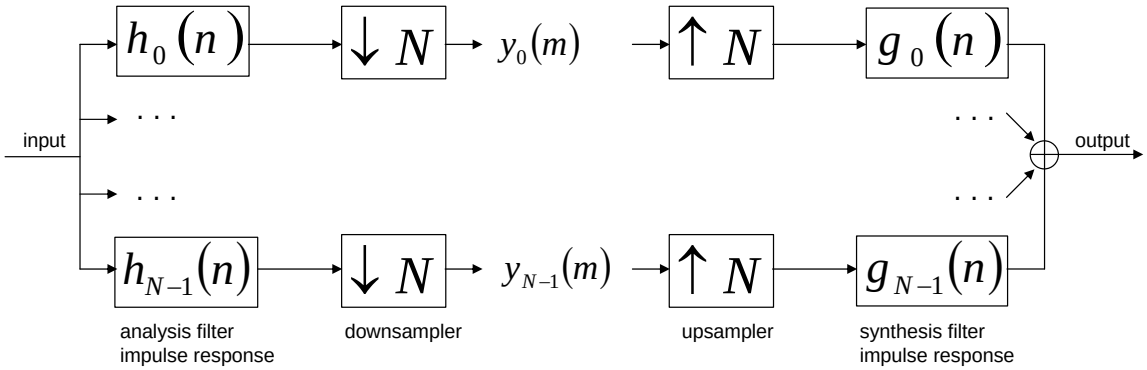

Figure 3.2: Basic principle of transform coding

shown in Figure 3.2. Furthermore, the mentioned audio coders exploit psychoacoustic effects using a psychoacoustic model.

The MP3 standard uses a hybrid analysis filterbank. In the encoder, the signal is first filtered by a so-called QMF filter bank with 32 subbands [138]. Next, each subband is then filtered by a MDCT filter bank with 6 or 18 subbands. The MDCT is an efficiently implementable modulated filter bank which consists of frequency shifted, i.e., modulated, versions of a low pass prototype filter. Depending on whether the signal is stationary or transient, one of the two different MDCT subband numbers is used (or switched to) [139], resulting in either 576 or 192 spectral lines in total, which corresponds to a switching ratio of 3:1. This switching enables the coder to adapt to a suitable time and frequency resolution depending on the signal characteristics. This limits the spreading of quantization errors over time after the reconstruction in the decoder, and should hence avoid so-called pre-echos, audible distortions in the reconstructed decoded signal before the attack or transient [136].

The psychoacoustic model estimates the current time- and frequency-dependent masking threshold of the input signal. The term masking in the frequency domain describes the fact that some parts of the spectrum are perceived by the ear with high precision or not at all in the presence of another signal at a nearby frequency (masker), which can be a narrowband noise or a tone. Hence neighboring sub band signals can have masking effects on each other resulting in frequency-dependent increases of the hearing threshold. Signals at frequencies with a level below this threshold cannot be perceived, i.e., they are masked, and can therefore be quantized to zero. If the quantization noise in each subband stays under the masking threshold, no difference between the original and compressed

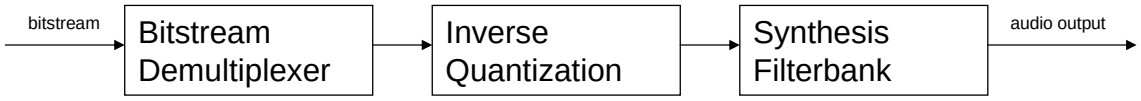

Figure 3.3: Basic audio decoder

signal is audible [136, 140].
In the quantization step of the audio coder, the output of the psycho-acoustic model, the masking threshold, is applied by adapting the quantization step size such that the quantization error ideally stays below the computed masking threshold for each subband. A number of 2- or 4-dimensional Huffman tables are employed to assign frequently occurring (smaller quantized) values to shorter code words, while less frequent values are assigned to longer code words. By varying the quantization step sizes, the bit rate is varied until it meets the requirements of the noise control loop [136]. Finally, all relevant information (i.e., the coded spectral values and additional side information) is multiplexed into a bitstream and transmitted to the decoder.
On the decoder side, all before mentioned steps are done inversely and in reversed order. This is shown in Figure 3.3. After demultiplexing the bitstream, inverse quantization is performed, followed by the synthesis filterbank. The filterbank transforms the signal from the time/frequency domain back to the time domain, using the inverse MDCT. This results in the reconstructed audio signal, which then can be played out.
The MPEG-2/-4 Advanced Audio Coding (AAC) standard is the follow-up standard to MPEG-1 Layer 3 and uses features such as temporal noise-shaping and long term prediction. Instead of the 2-stage hybrid filter bank of MP3, it uses a higher resolution filter bank, and MDCT with 1024 or 128 subbands — again switchable to avoid pre-echos. Temporal noise-shaping uses prediction across frequency to shape the quantization error in the time domain such that it roughly follows the signal power envelope. This results in improvements for time-varying signals, which do not vary quickly enough for switching the filter bank in the lower subband number mode, especially in speech or speech-like signals. AAC optionally has long term prediction, a technique that is commonly used in speech coding, which is based on the fact that stationary signals can be predicted to a certain extent. By only encoding the error between the predicted and the actual value, which should be considerably lower in amplitude, further bit rate can be saved. However, this requires higher computational complexity in the encoder, and hence is often not used [136].

3.3.2 Identification of the type and the parameters of the audio coder

The extraction of footprints from audio contents is in its infancy, restricted to individual processing steps and applications. For instance, state-of-the-art coding-based footprint detectors assume the usage of particular audio codecs [141, 142].
There are several reasons for the appearance of the coding-based footprints. They are implicitly inserted into audio signals by lossy coding due to *quantization*, which is usually performed in the subband domain, whose structure is characteristic for a class of audio coders. Hence, the quantization step size can be estimated by inverting the synthesis filter bank, e.g., using a corresponding analysis filter bank on the decoded audio stream. The estimation of the step size and the number of steps used in the quantization can be used to detect whether an audio stream was previously encoded at a lower bit rate before it was re-encoded at a higher bit rate (so called “fake quality” detection).
Another characteristic footprint is related to the *framing structure*. Audio coders produce the above mentioned subband decompositions on a frame based signal structure. If the correct framing

structure and correct filter bank is used, then the subband signals have an empirical distribution that consists only of just a few amplitude steps produced by the inverse quantizer. Thus, the size and the number of these amplitude steps are an indicator of the used bit rate, whereas the used filter bank is an indicator of the used coder.

The "Inverse Decoder"

The idea of the "inverse decoder" has been first discussed in [143]. Traditionally, audio decoding is considered as a one-way operation, aimed to retrieve an uncompressed audio format. Herre and Schug [143] pose the following questions:

- Is it possible to extract the compression parameters from an analysis of the decoded audio signal?
- Can a decoded audio signal be translated back into its bitstream representation, assuming a particular audio coder?

The problem of the translation of a decoded audio signal back to its bitstream representation is referred to as the *inverse decoding problem*. The authors address the generic audio coding structure and present the details of the "inverse decoder" algorithm based on the MPEG-2/4 AAC coder. The analysis of the encoding parameters is done in the following step-by-step process:

1. The decoder framing grid is determined.
2. For coding schemes with flexible coder filterbanks, the filterbank parameters are recovered. This enables to calculate an approximation of the spectral values inside the decoder.
3. The quantization information is estimated from the quantized spectral values.
4. Other parameters, such as joint stereo coding modes, can be recovered if applicable.

The experimental results show that the recovery of several basic encoding parameters is feasible.
An "inverse decoder" specialized on the popular MP3 coder is presented in [141]. Here the framing structure is determined by trying different frame-offsets and then detecting at which offset quantized amplitude distributions occur. Then the quantization step sizes, and many more parameters used in MP3 coding are estimated, with the goal of a complete "inverse decoder". The experimental results include the evaluation of the overall reconstruction precision of the system. Here the decompressed inverse decoded audio signal is compared to the decompressed original bitstream by means of the standardized PEAQ (Perceptual Evaluation of Audio Quality) measurement method.

"Fake Quality" Detection

Distribution of digital music over online stores has become extremely popular during the last years. The quality of distributed digital audio is one of the important issues. Here, low quality versions encoded with a low bitrate are often used to enable the preview for the customers. Unfortunately these reduced versions can become an object of a fraud. For example, a freely distributed low quality preview MP3 can be transcoded at a higher bitrate and sold online as a high quality product.
Recently, several publications addressed this issue. D'Alessandro and Shi [142] show the bitrate of MP3 files can be detected from the audio samples by analyzing high frequency components. They define five classes of bit rates (CBR 128 kbps, 192 kbps, 256 kbps, 320 kbps, and VBR-0) and apply a Support Vector Machine for the classification. In the experimental setup the average success rate reaches 97%.

Yang et al. [144] propose a method to recover the original bitrate and to detect fake high quality MP3 by analyzing the number of the MDCT coefficients with the small values. In particular they show that there are fewer MDCT coefficients of small values in fake-quality MP3 than in normal MP3. This findings are supported by the theoretical analysis on the quantization artifacts during the double-compression (first time with the low bitrate and second time with the high bitrate). The accuracy of the fake-quality detection method exceeds 97%. Unfortunately, it is not possible to compare these results with the ones in [142] as the experiments are carried out with different datasets.

Chapter 4

Editing

4.1 Image Editing

When we speak of image editing we mean any processing applied to the media that aims to change its content. There are many different reasons for modifying the content of an image: the objective could be, e.g., to improve its quality, or to change its semantic content. In the former case, the processed image will carry the same information as the original one, but in a more usable/pleasant way. Hence, we refer to this kind of editing as "benign". Conversely, in the latter case, the information conveyed by the image is changed, usually by adding or hiding something to alter its content. We refer to this kind of editing as "malicious".
Figure 4.1 provides a simple classification of editing operators, along with some illustrative examples for each identified class. We see that some operators are likely to be used only for benign editing, while others are clearly intended for malicious attacks. This is the case, for example, of Cut&Paste, which consists of splicing two or more images to create new content. In the middle, there are some operators (e.g. cropping) that can be employed either for slight post-production editing or for changing the carried information.

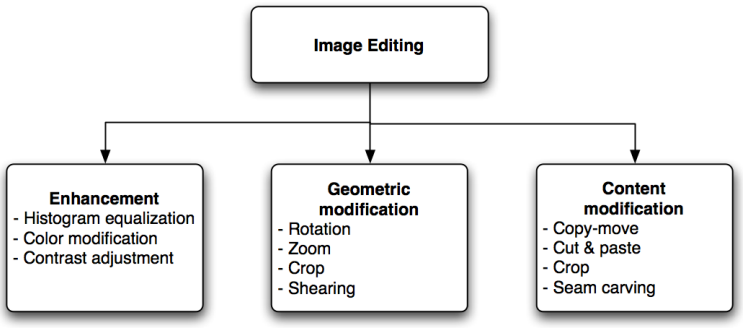

Figure 4.1: Main types of editing operators applicable to images.

In this section we survey several works that were proposed to detect various kinds of editing in images. Just a few works have been proposed for detecting benign editing (e.g. median filtering, contrast enhancement, scaling and rotation). Instead, a great deal of the research focused on methods

for detecting malicious editing. We identify two kinds of methods for creating forgeries using still images: copy-move attacks and cut-and-paste attacks.
Copy-move is one of the most studied forgery techniques: it consists in copying a portion of an image (of arbitrary size and shape) and pasting it in another location of the same image. Clearly, this technique is useful when the forger wants either to hide or duplicate something that is present in the original image.
Cut-and-paste is the other important image forgery technique: starting from two images, the attacker chooses a region of the first and pastes it on the second, usually to alter its content and meaning. This procedure is known as *splicing*. Splicing is probably more common than copy-move, because it is far more flexible and allows the creation of images with a very different content with respect to the original. This is demonstrated also by the huge amount of work on this topic.

4.1.1 Techniques for editing detection

In the following we will discuss forensics techniques that search for traces left by editing operators, without caring about the actual objective of the user who altered the image. From this point of view, detecting the use of a contrast enhancer is as important as finding that an image has been spliced.
All multimedia forensics techniques are based roughly on the same hypothesis: editing performed on the media leaves some kind of footprints in it. Therefore, by searching for such traces one might be able to detect the editing undergone by the media. In the next sections, state-of-the-art techniques will be grouped according to the kind of feature they look for. In turn, this indicates the kind of processing that can be detected.
We start by providing a basic distinction of traces left by editing operations. The first important distinction is between:

- Traces left at "signal level". In the course of processing, changes induced on the media leave some (usually invisible) footprints in its content.
- Inconsistencies left at "scene level". For example, shadows, lights, reflections, perspective and geometry of objects.

Furthermore, focusing on signal-level traces, different forms of editing can be revealed either because of the footprints left by the operations themselves (e.g. detecting periodicities in resampled images), or because they alter the footprints of previous processing steps (i.e. those introduced during acquisition or coding, as described in Sec. 2.1, 3.1 for images).
Clearly, inconsistencies at signal level and at scene level are somewhat complementary: a forgery that is perfect from the scene level point of view could be easily detectable using tools working at signal level and vice-versa. Furthermore it is clear that while tools working at signal level can detect non-malicious processing like contrast enhancement, tools working at scene level are unlikely to do so.

4.1.2 Signal processing based techniques

This section discusses methods that detect image editing by using signal processing tools designed to reveal footprints left during the editing phase. Examples of these footprints are: correlation between neighboring pixels, inconsistencies in JPEG compression artifacts, inconsistencies in device sensor noise, etc. Usually, one or more features are extracted and interpreted through a statistical model.

Copy-move detection

Copy-move attacks have been defined at the beginning of Sec. 4.1. Usually, it as assumed that the attacker will perform some modifications to the pasted pixels, in order to make them fit better with the surrounding content. These modifications include, for example, rotation, resizing, smoothing and contrast modification. Therefore, it is very important to be able to detect regions which have been modified before or after being pasted.

Several approaches to copy-move detection were presented by J. Fridrich in [145], namely: exhaustive search, correlation between the image and its cyclic-shifted versions, and robust match. While the first two approaches can be effective, they are computationally unaffordable. Robust match (which inspired the development of several other works) deserves more attention. Assuming that copied regions are connected components of reasonable size, the image is analyzed with overlapping blocks of small size: for each of them the DCT is calculated, and quantized coefficients are stored in an array. The arrays corresponding to different blocks are then lexicographically ordered. Now, if a region has been copied, a high number of blocks will present a matching DCT footprint and the *shift vector* connecting one block to its copy will be the same for all of them. Counting the number of rows in the list which satisfy these two assumptions, the authors successfully revealed tampering on three sample images provided in the paper, with a small amount of false alarms (mostly due to homogeneous regions). Notice that lexicographically ordering the features arrays highly simplifies the identification of identical rows: they are easily searched by going through all the rows of the ordered matrix and looking for two consecutive rows that are identical. The method is robust to noise addition, strong JPEG compression, and small amount of resizing (up to 10%) and rotation (up to 5°).

The idea of extracting footprints from blocks and then use lexicographic ordering has been exploited in other works. In [146], the signature of each block is obtained by using color-related features, yielding an high tolerance to noise addition and JPEG compression. In [147], Popescu et al. propose to use a Principal Component Analysis instead of quantized DCT coefficients as block signature. This allows a more compact representation of each block (which speeds up search and lower memory consumption) and improves the robustness to compression and noise addition. However, rotation and zooming are not tolerated. Later, Bayram et al. [148] introduced the use of Fourier-Mellin Transform (FMT) as block signature. Since FMT is invariant to rotation and scaling, the authors claim that their method is capable of detecting pasted regions even if they are rotated or scaled. Indeed, detecting performance for scaled/rotated images is not far from that of [145] (scaling up to 10% and rotation up to 10° are tolerated). Along the same line, Wu at al. [149] recently proposed the use of Log-Polar Fourier Transform as signature to yield invariance to rotation and scaling. Results reported in the paper show a significant improvement, although only on a few sample images: detection is successful even when the pasted region is rotated by 70° or scaled by 60%.

Huang et al. introduced a completely different approach [150], which is based on SIFT (Scale-Invariant Feature Transform) local features. The basic concept is to use SIFT descriptors [151] to find matching regions within the same image. The main difficulties are choosing an appropriate matching strategy and properly partitioning image keypoints into subsets (in order to search for matchings between their elements). Authors propose to use a Best-Bin-First [152] matching strategy, and to recursively split the set of keypoints into two subsets. Matches are identified across subsets, and the algorithm stops only when each keypoint has been matched to another keypoint. Experimental results show that this forgery detection technique effectively inherits the robustness features from SIFT: strong rotation, scaling, compression and noise addition are tolerated.

The idea of using SIFT has been later exploited in two recent works. Amerini et al [153], used SIFT

keypoints to detect the forgery and to estimate the geometric transformation performed during the tampering process. The algorithm extracts SIFT descriptors for each keypoint, evaluates descriptor matchings, applies a clustering algorithm to identify suspect regions and, finally, estimates the geometric transformation between the two regions. Experiments carried on a publicly available dataset (`http://www.ee.columbia.edu/ln/dvmm/downloads/PIM_PRCG_dataset/`) of 100 forged images show very good performance when detecting copy-move forgeries, even when the copied region is rotated up to 90° and scaled up to 50%.
A very similar approach has been independently developed in [154]. In contrast to [153], this work is more focused on finding the pasted region contours (which were not provided in [153]) rather than estimating the geometrical transformation.
The development of SIFT based techniques did not inhibit researchers from continuing investigating block-based approaches. Recently, Ryu et al. [155] proposed an algorithm in which the signature for each block consists of Zernike moments [156], which are invariant to rotation. Although this method improves the performance of [153] and [154] when dealing with rotated regions, it suffers from scaling and other forms of affine transformation. Another block-based approach has been published by Bravo-Solorio et al. [157]. This approach relies on color-based signatures and proves to be robust to rotation but not to substantial scaling.
Although copy-move forgeries have already received a lot of attention and inspired a large number of papers, the detection of this kind of attack remains a challenging problem. Indeed, many open issues are still to be explored such as, for example: understanding which is the original patch, between two copies; improving performance in detecting small copied regions; making detection techniques more content-independent (up to now, attacks on very smooth regions, e.g. depicting the sky, are usually considered false positives).

Splicing detection based on camera artifacts

When acquiring an image, each camera leaves various kinds of footprints in the signal (see Sec. 2.1). If two images are taken from different cameras, they will probably exhibit different footprints. Therefore, splicing might result in the introduction of inconsistencies in these slight traces. We present some works which reveal inconsistencies in camera artifacts to detect splicing.
One of the first works based on this idea is due to Chen et al [6]. It exploits sensor noise (namely, photo-response non-uniformity, PRNU) to reveal both image origin and integrity. If the forensic analyst has a small amount of images taken from the same camera, he can estimate the PRNU mask for that device and use it to understand if a newly received image comes from that device. Moreover, this technique can also be used to reveal if a part of the image does not come from the expected device. Indeed, if a portion of an image taken with a camera is replaced with another taken from a different device, the PRNU mask in that region will be inconsistent with the one of the original camera. Thus, a two-hypothesis (tampered/non-tampered with) test can be performed block-wise over the image, in order to locally assess its integrity and to reveal the position of regions that have been tampered with. Well documented experiments in [6] show that this method is effective (true-positive rate for tampered pixels around 85%, false positive around 5%) provided that the examined image has not undergone heavy lossy compression. However, performance is still good when the image is compressed using JPEG at a quality factor greater than 75.
Along with PRNU, another important artifact left by cameras during acquisition is that due to Color Filter Array demosaicking (see corresponding section in Chapter 2). Popescu et al. [14] proposed a technique for detecting CFA interpolation in an image. The algorithm assumes a linear interpolation kernel, a simplistic but effective hypothesis compared to complex methods adopted in commercial devices, and uses an Expectation-Maximization (EM) algorithm to estimate its parameters (i.e.

filter coefficients). The method determines a p-map, which gives for each pixel the probability of being correlated to surrounding pixels, according to the currently estimated kernel. Depending on the actual CFA pattern, some pixels are interpolated whereas others are directly acquired. Hence, the correlation map exhibits a periodic behavior, which is clearly visible in the Fourier domain. In the presence of splicing, the CFA pattern is violated and the periodicity of the map compromised. Thus, by analyzing the local periodicity of the p-map, the authors are able to understand whether a part of the image is original or not. This approach is less robust to JPEG compression compared with the one based on PRNU, but is characterized by a lower false-positive rate.
Dirik et al [158] also exploit CFA interpolation artifacts for determining image integrity. They propose two methods for checking the presence of demosaicking artifacts. The first consists in estimating the CFA pattern of the source digital camera. The image is simply re-interpolated assuming many different patterns, and the pattern which leads to the smallest mean square error is chosen. The second method leverages the low-pass nature of common demosaicking kernels, which is expected to suppress the variance of underlying PRNU noise. Therefore, the presence of demosaicking artifacts is detected by comparing the variance of sensor noise in interpolated pixels against the one in directly acquired pixels.
In [159] Hsu et al. explore the usage of another kind of camera artifact, i.e. the Camera Response Function (CRF), which maps in a non-linear fashion scene irradiance to image brightness. The basic idea is to look for inconsistencies in the artifacts. The image is automatically segmented, the CRF is estimated on locally planar irradiance points (LPIPs) near to region borders, and a comparison among the estimated functions for distinct regions sharing the same border is performed. Various statistics of these errors are used as features for training an SVM classifier. Classifications results for various region borders are fused to get a final judgment about image integrity. Results achieve 90% recall with 70% precision, but these values are obtained on a challenging real-world dataset.

Splicing detection based on coding artifacts

The methods in Sec. 4.1.2 assume that the original image contains camera artifacts. As a matter of fact, most images are stored in a lossy compressed format, typically JPEG. Unfortunately, JPEG erases almost completely the slight artifacts introduced by cameras, even when images are compressed at a reasonably large quality factor (e.g. 80). As a matter of fact, the aforementioned methods cannot be trusted when working on JPEG-compressed images.
On the other hand, JPEG compression introduces other characteristic artifacts, e.g. the well known "blocking-artifacts". Therefore, several methods have been proposed to assess image integrity by looking for inconsistencies in these traces. When a forgery is created starting from JPEG images, many different settings are possible. We can identify two main situations:

- Single Compression Forgery (SCF): an original JPEG image, after a localized forgery, is saved again in JPEG format without resizing. DCT coefficients of unmodified areas will undergo double JPEG compression thus exhibiting double quantization (DQ) artifacts. Conversely, DCT coefficients of forged areas will show the presence of only the last compression step and will not present DQ artifacts;

- Double Compression Forgery (DCF): in this kind of forgery, it is assumed that a region from a JPEG image is pasted on a host image (which is not JPEG compressed) and that the resulting image is JPEG compressed. Depending on the placement of the forged region, the forged region may exhibit Not Aligned (NA) JPEG compression artifacts (i.e. it has been compressed twice, but compression grids are not aligned) or Aligned-JPEG compression artifacts (i.e. the pasted

region is compressed twice). Assuming a random positioning, the probability that the forged region will not be aligned is equal to 63/64.

The vast majority of proposed algorithms for splicing detection based on JPEG artifacts focus only on *one* of the possible tampering scenarios outlined above. This is the case, e.g., of the work by Luo et al. [160], in which a DCF with NA compression setting is assumed. The method is based on visual, spatial and blocking artifacts exhibited by JPEG-compressed images. In this scenario, the original part of the image exhibits regular blocking artifacts, while the pasted one does not, since the second compression was not aligned with the first. The method extracts some features, cumulated over the whole image, which are fed to a classifier in order to distinguish regions in which blocking artifacts are present from those in which they are not. If the suspected region (which is known by hypothesis) does not exhibit blocking artifacts, then it is classified as tampered. Results are good only when the quality factor of the last compression is much higher than the one used for the first. Furthermore, the method is reliable only when the tampered region is very large, i.e. above 500x500 pixels.
The hypothesis of knowing in advance the suspect region, although being very strong, is common to many works. When the region is unknown, one possible work-around is to check the integrity of the image block-wise. This approach is pursued in [160], and it is computationally demanding. An extension of this work has been proposed by Barni et al. [111]. Instead of checking for all possible blocks, the authors suggest to segment the image and then apply the method in [160] to asses the integrity of each segment. This is reasonable, since cut-and-paste tampering usually introduces a well defined new object in the image, which is likely to be isolated by segmentation.
Ye et al. [100] provided a different way to analyze blocking artifact inconsistencies, which is based on the observation that the histogram of DCT coefficients concentrates only on multiples of the quantization step. Thus, analyzing the power spectrum of DCT coefficients, the quantization table can be estimated and used to check if every part of the image is quantized consistently.
Another approach covering the DCF-NA scenario is proposed in [112]. There, the mixing of compression artifacts in forged regions is considered as a noisy convolutive mixing model, and a solution for the model is obtained by using an approach based on Independent Component Analysis. Tampering identification is still performed by means of a classifier. Results are improved with respect to [160] by 5%, especially when tampered regions are small.
A recent work addressing the DCF-NA scenario is the one proposed by Bianchi et al. [161], which does not rely on any classifier. Instead, a simple threshold detector is employed. The main idea behind the method is that of detecting NA-JPEG compression by measuring how DCT coefficients cluster around a given lattice (defined by the JPEG quantization table) for any possible grid shift. When NA-JPEG is detected, the parameters of the lattice also give the primary quantization table. Results obtained in this work show an improvement with respect to [160] and [112]: a forged region of 256x256 pixels is sufficient to equal the best results presented in previous works, and good performance (over 90%) is obtained even in the presence of similar first and second quantization factors. Consequently, this method retains good performances even when the last quantization is coarse, e.g. corresponding to a quality factor equal to 70.
Several techniques have been proposed in the literature targeting the SCF scenario. Farid [162] detects forgeries by recompressing the image at several quantization factors and looking for "ghost" effects, which reveal that the coefficients were previously quantized with larger quantization step size (i.e. at lower quality). The method can detect very small tampered regions, but requires the suspect region to be known in advance.
Another approach is pursued by Lin et al. [163], which were the first ones to exploit the detection of double quantization effect in DCT coefficients to tackle with the problem of tampering detection.

In the SCF scenario, if a portion of the image does *not* show DQ effect while the remaining part does, then it is likely to be tampered with. Surprisingly, results reported in [163] are not very good (mean detection rate under 70%). A more recent work from Bianchi et al. [164] exploits the DQ effect in a more effective way: DCT coefficient histograms are modeled considering that coefficients of tampered images will have statistics coming from two different models. This method does not rely on a classifier and produces as output, similarly to [163], a fine-grained map of tampering probabilities.
Recently, Chen et al. [165] have proposed a technique which is, in principle, capable of detecting both aligned and misaligned double compression. The method combines periodic features in spatial and frequency domains, taking into account both the visual blocking artifacts and the periodicity of DCT coefficients. Results show an improvement with respect to [160], but are not compared to other works for the SCF scenario.

Splicing detection based on general intrinsic footprints

The works described so far are able to detect tampering by leveraging specific footprints left or erased during processing. Conversely, the works described in this section are more focused on finding footprints left in the signal during tampering without considering the phenomena that caused the presence of these effects. The key idea in these works is that spliced images bring anomalies in the image statistics, which make them distinguishable from the original ones. This kind of approach usually allows to detect many different kinds of tampering at the price of lower accuracy.
One of the first approaches in this direction was proposed by Avcibas et al. [166]. In this work the emphasis is on finding features of the image which reflect the presence of image manipulations, while being independent from the content. The authors select four image quality metrics (taken from a previous work on steganography) and define new correlation coefficients between the suspected image and the reference original, to which various kinds of processing are applied. They feed all these features to a linear regression classifier. Results are below average (accuracy between 70 and 80%) compared to nowadays techniques, but were really encouraging when the paper was published.
Chen et al. [167] employ a classifier, fed with three categories of features: statistical moments of the Characteristic Function (CF) of the image, moments of the wavelet transform of the CF, and low order statistics of the 2D-phase congruency. Accuracy, taken on a well known splicing dataset, is on the average still below 85%.
Similarly, Shi et al. [168] use a classifier trained with statistical moments of the image itself, of the DCT of the image (performed block-wise with various block dimensions), and statistical moments of the LL sub-band of the wavelet transform. Performances are better than in previous works, reaching a level of accuracy around 90%.
A comprehensive approach has been developed by Swaminathan et al. [16]. In this work, intrinsic footprints of the in-camera processing operations are estimated through a detailed imaging model and its component analysis. Further processing applied to the image is modeled as a manipulation filter, for which a blind deconvolution technique is applied to obtain a linear time-invariant approximation and to estimate the intrinsic footprints associated with these post-camera operations. If the estimated post-camera operations are far from being identity functions, the image is classified as tampered. The method can be used for several forensic tasks: steganalysis, filtering detection, tampering detection. On the other hand, reported accuracy values are not very high.
Another approach which aims at detecting multiple kinds of processing is introduced by Stamm et al. in [169]. First, the histograms of original images are properly modeled. Then, the effect of many different operations (e.g. contrast enhancement, histogram equalization, noise addition) on the histogram is studied. In particular, contrast enhancement can be detected locally, allowing

tampering detection.

Resampling detection

Images are often scaled (often referred to as zooming/shrinking) and/or rotated. Scaling and rotation operate in the domain of the signal, in that they affect the position of samples but not their amplitude value. While an image can be scaled or rotated by an arbitrary amount, the support of the destination image remains discrete, so the original image must be *resampled* to a new sampling lattice. Resampling introduces specific correlations in the image, which can be used as evidence of editing. Furthermore, when composing two or more images, it is often necessary to adapt the spliced region by rotating and scaling it. If the traces left during this process are not destroyed by compression or further manipulations, they can be exploited to reveal tampering. For this reason, resampling detection techniques can be exploited for detecting both benign editing (for example, scaling or rotation of the whole image) and malicious editing (by checking if only a certain region has been resized, thus altering the information carried by the image). As such, in describing available techniques for resampling detection, we will point out when they can be used for tampering detection as well.

Popescu et al. [170] proposed a way to detect periodic correlations introduced in the image by common resampling kernels. The method is very similar to the one introduced by the same authors in [14]. The interpolating kernel is estimated using an Expectation-Maximization algorithm, and the probability that a pixel has been resampled is obtained. This map can be used to search for tampered regions. Accuracy of the method is very high, provided the image has not been compressed. In fact, even very slight JPEG compression (e.g. quality factor above 95) is sufficient to erase resampling artifacts.

Another approach to resampling detection has been developed by Mahdian et al. [171]. They study and analytically describe the periodic properties of the covariance structure of interpolated signals and their derivatives. The core of the method is a Radon transform applied to the derivative of the investigated signal, followed by a search for periodicity. Results are similar to those obtained in [170], but the authors claim that their algorithm is faster and not parameterized. In [172] Dalgaard et al. explore the importance of differentiation (which is used, for example, in [171]) or similar pre-filtering operations in resampling detection.

Another interesting approach to resampling detection is the one proposed by Kirchner et al. [173]. The authors show how the variance of prediction residuals of a resampled signal can be used to describe periodic artifacts in the corresponding p-map (defined by Popescu in [170]). By recognizing that the formation of periodic artifacts does not depend on the actual prediction weights, they proposed a simplified detector. The exhaustive search in a set of candidate transformations is replaced with a much faster search for anomalies in the p-maps cumulative periodogram. Reported experimental results on a large image database show that this detector is orders of magnitudes faster than the scheme in [170], while achieving similar performance.

Enhancement detection

Creating credible image forgeries is getting simpler. At the same time, it is becoming more difficult to find images which are published without having undergone at least some benign enhancement steps. Although not aiming at modifying the "semantic" content of an image, enhancement techniques (e.g. smoothing, contrast enhancement, histogram equalization, etc.) are useful to improve the perceived quality. Surprisingly, enhancement detection has received much less attention than forgery detection so far.

Median filtering is a quite commonly used smoothing technique. An interesting approach to the detection of median filtering has been proposed by Kirchner in [174]. The basic idea is that median filtered images exhibit so called "streaking artifacts" (pixels in adjacent rows or columns share the same value). These artifacts can be analyzed by considering first order differences for groups of two pixels and then studying the corresponding histograms. This simple approach yields extremely high detection rates, provided that images are not compressed.
Several works have been proposed by Stamm and Liu, aiming at detecting and estimating contrast enhancement and histogram equalization in digital images. The first of these works targets the detection of the operator [175], while in [176] an extension is provided in order to estimate the actual mapping induced by the contrast enhancement operator. In both cases, the key idea is to reveal footprints left in the image by the operator, which consist in the formation of sudden peaks and zeros in the histogram of pixel values. These techniques were originally thought for enhancement detection, but they have also been successfully applied to splicing localization in a previously cited work [169] by the same authors.

Seam carving detection

Until a few years ago, there were only two ways for reducing the size of an image: cropping part of the content at the borders or scaling the whole image (or, of course, a combination of these two methods). However, cropping is not a good option when borders contain relevant information, while scaling forces a reduction in image resolution, lowering the quality of the media. A very interesting and recent solution to this problem is *seam carving* [177]. The basic idea is to automatically detect, if any, paths of pixels (seams) of the image along which no relevant content is present. If detected, these paths are eliminated and the image size is reduced. We may think of this technique as a sort of content-dependent cropping.
Although aiming at preserving original content, seam carving remains an editing operator. Two works have been proposed to detect if an image has undergone this kind of processing, respectively by Sarkar et al. [178] and Fillion et al. [179]. In [178] changes in pixel values near the removed seams are searched by building a Markov model for the co-occurrence matrix in the pixel and frequency domain and used as features to train a classifier. In the same work, an approach based on Popescu and Farid's EM method in [170] is introduced for detecting seam insertion (a complementary operation with respect to carving performed to enlarge an image). Experimental results show that performance strongly depends on the training phase. Using the most general training set, accuracy is approximately 84%. Results are also provided for rotation and scaling detection, but they are not as good as those obtained by specifically targeted methods.
The work from Fillion et al. [179] also relies on a classifier, but the chosen features are very different. Basically, three features are employed: the first takes into account how energy is distributed in the image histogram; the second exploits the fact that applying a second seam carving to an image reveals if low energy seams have already been removed; and the third is based on statistical moments of the wavelet transform. This work also targets the detection of both seam carving and insertion. Performance is evaluated on two separated datasets: the first contains images on which "benign" use of seam carving has been made, while the second contains images on which seam carving has been used for removing objects. Performance is higher on the former dataset (ranging from 70 to 90% depending of the amount of reduction), while accuracy drops significantly on deliberate tampered images, showing the need for further research in this area.

4.1.3 Geometry / physics based techniques

Up to now we have presented only works that tackle editing detection from a signal processing point of view. Inconsistencies or footprints left by processing operations are searched using statistical tools and models. In this section we introduce a different kind of approach, usually referred to as "geometry / physics based". Instead of looking at signal properties, the idea is to reveal inconsistencies introduced by tampering at the "scene" level (e.g. inconsistencies in lighting, shadows, colors, perspective, etc.). One of the main advantages of these techniques is that, being fairly independent on low-level characteristics of images, they are extremely robust to compression, and remain applicable even when the quality is low. Going to extremes, these techniques can be applied even for old analog photographs: for example, Farid carefully examined a famous photo of Lee H. Oswald[1] that dates back to 1963 [180]. To the best of our knowledge, methods of this kind have only been developed to target forgery detection.
The basic consideration underlying these techniques is that human brain is not good at all in creating forgeries that are consistent from a geometric/physic point of view. This leads to think that most forgeries will likely contain slight errors that can be detected. However, humans are not good at detecting these kinds of inconsistencies. As a consequence of these facts, the great majority of the works in this category provide tools for helping a human analyst to search for inconsistencies at the scene level. Therefore, it is usually assumed that the analysis is *assisted*. Furthermore, some kind of physical footprints are left by the acquisition device (like chromatic aberration, see later) and do not depend on the scene content but on the device characteristics.
Notice that the kind of inconsistencies searched by these methods are likely to be introduced when a cut-and-paste attack is performed. Conversely, a copy-move attack is usually hard to reveal especially when targeted to hide something, except for a few techniques (for example, those based on lens distortion). Similarly to the previous section, we adopt a simple classification based on the kind of footprint. Notice that it is not easy to objectively assess the performance of these techniques because, being human-assisted, they can not be tested on massive amounts of data. As a consequence, while each paper shows very good results on all of the reported examples, the validity of the proposed methods in different contexts is not easy to predict.

Splicing detection based on Lighting/Shadows

One of the most common errors when creating a forgery is to neglect how objects in the scene interact with light. Cutting an object from a photo and pasting it into another requires to adapt object illumination, and to introduce consistent shadows in the scene. When this is not done, inconsistencies in lighting and shadows can reveal that the forged image is not real.
Johnson and Farid [26] examined how light inconsistencies can be detected in a standard 2D image. The task is not trivial: complex lighting environments (multiple light sources, diffuse lighting, directional lighting) give rise to complex and subtle lighting gradients and shading effects in the image. However, the authors take some simplifying hypothesis (i.e. infinitely distant light source, Lambertian surfaces, etc.) under which a nine-dimensional model is sufficient to describe mathematically the illumination of the scene. Then, they show how to approximate a simplified 5-D version of this model from a single image, and how to stabilize the model estimation in the presence of noise. Inconsistencies in the lighting model across an image are then used as evidence of tampering.
Another, less general, idea is presented in [27] by the same authors. Here spotlight reflections in human eyes are exploited to check if two persons in the same image have been actually taken from

[1] Lee Harvey Oswald (October 18, 1939 - November 24, 1963) was, according to four government investigations, the sniper who killed John F. Kennedy on November 22, 1963.

different photos. The approach to the problem is still mathematical: the geometrical transformation from world to image coordinates is estimated by letting the user select the limb of the eye (which is always circular, but is elliptic in the image). Then the normal to the eye surface is calculated and the direction of incident light is obtained with a very good approximation (precision of almost 5°). If light directions are significantly different for different subjects, splicing is revealed.
Riess et al. [24] propose a different approach to lighting-based tampering detection, based on illuminant color consistency over the image. The image is first segmented in regions of similar color. A user selects suspect regions among these, and a map is generated which shows how much each region is illuminated consistently with respect to the dominant illuminant colors. This map may also allow to understand if the image has been taken using a flashlight, or to understand the color of the light that illuminated the scene.
As stated before, inconsistencies in shadows are a good indicator of tampering. In [181], Zhang et al. proposed two methods to detect inconsistencies in shadows. The first method is based on shadow geometry, using a planar homology to check consistences of shadows size and directions. The second exploits shadow photometry, specifically shadows matte values, which often turn out to be useful in discriminating pasted shadows from original ones. This is reasonable, since changes in the color of shadows are most likely due to changes in scene illumination.

Splicing detection based on lens distortion

As described in Chapter 2, camera lenses introduce slight artifacts in the acquired images. We report two important works in this field: the first one, by Johnson et al. [182], exploits lateral chromatic aberration (LCA). The second, by Yerushalmy et al. [183], is mostly based on a type of artifact known as Purple Fringing Aberration (PFA). It is worth noting that both these methods do not require supervision. Chromatic aberration is caused by the fact that long (red), middle (green) and short (blue) wavelengths are not focused by the lens at the same point in the image plane, when the source light is off the optical axis. It is mostly visible along boundaries of objects that are distant from the center of the image. Purple Fringing Aberration, although having a much more complex origin (see [183] for details) is stronger and more visible (in the form of a blue-purple halo near the edges of objects in the image). In [182], the LCA effect is mathematically modeled. Then, parameters of the model which would correct errors in color channel phase are estimated block-wise along the image, generating various "distortion maps". Inconsistencies in these maps are easy to find and prove that tampering has occurred. The method proposed in [183] is perhaps simpler. Borders showing PFA effect are automatically selected. Them, the direction and the magnitude of the effect on each border is estimated and represented as a bidimensional vector in image coordinates, which must point to the center of the image for non-tampered images. If some vectors do not point towards the center, there is a high probability that the object has been pasted from another image.

Splicing detection based on inconsistencies in Geometry/Perspective

As stated before, the human brain is definitely not good at evaluating the geometrical consistency of a scene. Starting from this observation, some works have been developed to evaluate inconsistencies in the geometrical setting of an image. Of course, this problem is ill-conditioned because of the mapping from 3D coordinates to image coordinates during acquisition. Nevertheless, in simplified contexts, some kind of analyses can be performed.
As a first example in this class, we mention the work by Conotter et al. [184], which focuses on detecting inconsistencies in signs and billboards writings perspective. When a sign or a billboard is present in an image, it usually shows some writings arranged on a planar surface. This, together with a careful estimation of the character type which is used in writings, allows to estimate the

planar homography for that surface, which is compared to the one extracted from the image using, e.g., other planar objects present in the image. If the transformations are not consistent, it is highly probable that the writing is fake. The method works very well even when the attacker tries to correct writing perspective according to the destination image, for example by using common editing software.
Another interesting approach has been proposed by Kakar et al. in [185]. The method is based on discrepancies in motion blur in the image, usually caused by the slow speed of the camera shutter relative to the object being imaged. The proposed algorithm employes a blur estimation via spectral characteristics of image gradients, which can detect small inconsistencies in motion blur. Notice that the work tackles also with the case in which the attacker tries to artificially mimic motion blur, being able to distinguish natural blur from artificial one. The output of the algorithm is the image segmented according to consistent blur detected during the analysis, which can be manually analyzed to detect suspect inconsistencies.

4.2 Video Editing

In this section we introduce multimedia forensics techniques that have been developed to detect video editing and copying/reproduction.
Figure 4.2, which is an extension of Figure 4.1, shows that editing operators for videos can be readily split in two categories: intra-frame editing, in which a (group of) frames is modified independently from surrounding ones, and inter-frame editing, in which the sequence of frames is altered in a context-aware manner. Naturally, operators for intra-frame editing coincide with those defined for still images (each frame being an image). Conversely, inter-frame editing involves both the insertion or deletion of (a group of) frames and the editing of one or more frames performed by considering the content of adjacent frames. Similarly to the case of still images, editing can be performed for a malicious or benign purpose.

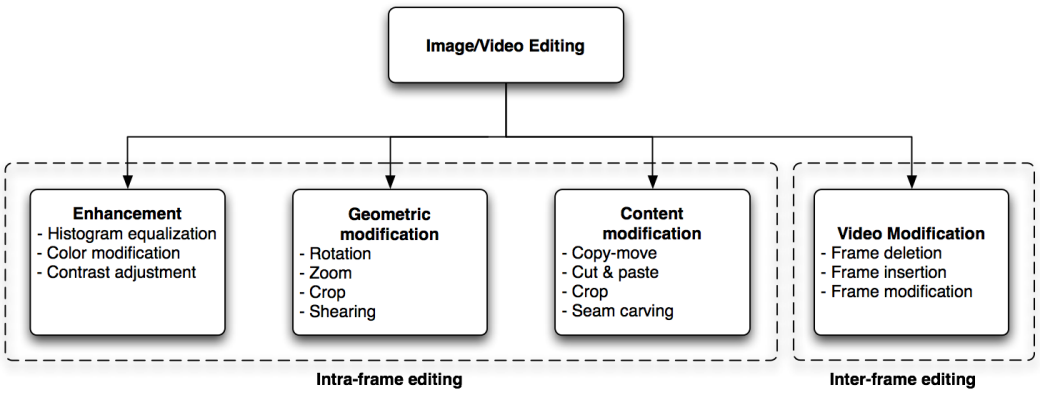

Figure 4.2: Main types of editing operators that can be used on videos. Notice that, the only difference between video and image editing is the possibility/necessity to consider groups of frames when editing a video (inter-frame editing).

The problem of detecting malicious editing in video is considered as a difficult task, mainly because of the huge amount of data that must be analyzed. Some techniques have been proposed, and will be presented separately according to the kind of features they rely on. While creating a fake

video is still not as easy as splicing two images, illegal duplication of videos is a problem for many real application scenarios. Video copy detection is usually targeted with techniques different from those employed in multimedia forensics, being similar to an information retrieval problem. However, video copy detection is very relevant to forensics, and a few works address the problem from this perspective.
All the methods described below exploit exclusively information coming from visual data, while audio is not considered.

4.2.1 Signal processing based techniques

Algorithms presented in this section target video editing detection from the signal processing standpoint. Imperceptible inconsistencies are searched for through statistical analysis and used to recognize extraneous phenomena which expose doctoring.
In the case of video content, many kinds of malicious editing operations are possible. A simple copy-move attack consists in replacing a (group of) frames with other ones, perhaps in order to conceal something. More complex doctoring attacks, instead, may be targeted to introduce something new in the video, or to change the appearance of something that is present.

Copy-move

Copy-move attacks are defined for video both as intra and inter-frame techniques. An intra-frame copy-move attack is conceptually identical to the one described for still images, and consists in replicating a portion of the frame in the frame itself. An inter-frame copy-move, instead, usually consists in replacing some frames with a copy of previous ones, with the aim of hiding something that entered the scene in the original video. To this end, partial inter-frame attacks can be defined, in which only a portion of a group of frames is substituted with the same part coming from a selected frame.
To the best of our knowledge, there is only one work authored by Wang and Farid [186] that targets directly copy-move detection in video. The method uses a kind of divide-et-impera approach: the whole video is split in subparts and a time and spatial "correlation matrix" for each subpart is calculated by evaluating the correlation between pairs of frames. These correlation matrices are used as fingerprints for each subsequence, and used to check if they are too similar and therefore suspect. The approach is defined both for still-camera and moving-camera scenarios. In the same work, a method for detecting region duplication, both for the inter-frame and intra-frame case, is defined. The method is based on a two-step analysis: first, suspect regions are searched using the well known Phase Correlation analysis [187]. Then, identified regions are further examined using correlation analysis. Results are good (accuracy above 90%) for a stationary camera, and still interesting for a moving camera setting (approx. accuracy 80%). MPEG compression does not hinder performance.

Doctoring detection based on camera artifacts

As explained in Chapter 2, the device used for acquiring data usually leaves some kind of footprint. This is also true for camcorders, thus multimedia forensics researchers explored how these artifacts can be used to detect forgeries in video.
A first result is the one from Kobayashy et al. [52]. In this work, the authors propose an approach to detect suspicious regions in video recorded from a static scene by using noise characteristics of the acquisition device. Specifically, photon shot noise is exploited, since it dominates other noise sources. Assuming that a static scene is recorded, this kind of noise can be modeled by considering the relation

between the mean and the variance of the observed pixel values in time. A characteristic Noise Level Function (NLF) can be estimated for the specific video. Using the estimated NLF as reference, pixel value statistics can be classified as consistent or inconsistent, thus allowing the detection of tampered regions. Reported experiments show good performance for the method (accuracy around 90%), but the strong hypothesis of a static scene limits the applicability of the technique.
Also the work by Hsu et al. [188] exploits noise residue for detecting forgeries in videos. Here, the PRNU noise pattern of the device sensor is estimated by subtracting from each frame its de-noised version. Frames are then divided in blocks and correlation of noise pattern between corresponding blocks in adjacent frames calculated. If extraneous frames are added to the video, the correlation value will drop, revealing the forgery. Of course, an accurate modeling of temporal noise correlation values of video blocks in forged and normal regions is needed: authors propose a Gaussian Mixture Model, whose parameters are estimated using an Expectation Maximization algorithm. Unfortunately, strong compression has a deep impact on PRNU, making this technique reliable only on good quality videos. Low bandwidth video, including most of those available on the internet, cannot be analyzed due to quantization noise.

Doctoring detection based on coding artifacts

As already stated, quantization noise introduced by compression causes the performance of camera-based forensics techniques to drop dramatically. Fortunately, similarly to the case of still images, artifacts due to compression can be used as footprints as well, allowing to conduct forensics analysis on highly compressed videos.
The first approach in this direction was from Wang and Farid [104]. They started by considering that a forged MPEG video will almost surely undergo two compressions, the first being performed when video was created and the second when video is re-saved after being tampered with. Therefore, depending on GOP (Group of Pictures) structure, some frames will undergo double JPEG compression, which can be revealed using the techniques described in Section 4.1.2. Furthermore, if a group of frames is either removed or altered, a desynchronization will occur in the GOP pattern that follows the edited part. This desynchronization can be revealed because motion compensation errors will increase. In [104], experiments are reported on just a few videos, but the method seems to be applicable in a general setting, provided that the suspect video is MPEG. Another work from the same authors [134] provides a more accurate analytical description of double compression, which allows to detect localized tampering up to a minimum of 16x16 pixels, assuming MPEG compressed video.
Wang and Farid proposed another approach [189] for detecting tampering in interlaced and de-interlaced video. For de-interlaced video, the authors explicitly model and estimate the correlations introduced by de-interlacing algorithms. They show how tampering can destroy these correlations. For interlaced video, they measure the inter-field and inter-frame motions, which are the same in the case of authentic video, but might be different in the case of forged video. Both techniques allow the localization of tampering both in time and in space. Furthermore, both algorithms can be adapted to detect frame rate conversion. Since compression partially removes inter-pixel correlations this approach is mostly suited for medium/high quality video. For interlaced video, instead, compression does not seem to hinder performance.

Copy detection

Video copy detection consists in identifying the duplicated and modified copies of a video clip among a large number of sequences. This is required to accomplish various tasks involved in identifying, searching and retrieving videos from a database. The most common approach in video copy detection

is to extract unique features from the visual content that do not depend on the device used to capture the video. However, in [190] Bayram et al. pointed out that robust content-based signatures may hinder the capability of distinguishing between videos which are not copies of each other, although sharing similar content. For this purpose, they proposed to use source device characteristics extracted from a video to construct a new video copy detection technique. In this scheme, rather than extracting a content-based signature from a video, a combination of the footprints of camcorders involved in the generation of a video are used as the video signature. Specifically, for each device the footprint is obtained by estimating the PRNU noise of the sensor. One may think that estimating sensor noise from a video is simpler than from a set of images. Unfortunately, this is not true due to several factors: frame sizes are smaller, decreasing the available information needed for reliable detection; successive frames are very similar, so averaging successive instances of PRNU noise patterns do not effectively eliminate content dependency; and finally, motion compensation may cause the loss of PRNU noise in some parts of the frames. However, having approximately 10 minutes of video, a good estimate of the footprint can be obtained, although non totally independent from the content. The resulting method proves to be very effective in detecting copied video, and, being based on features that are both dependent on content and device, has low false alarm probabilities even for videos with very similar content.

4.2.2 Geometry / physics based techniques

As already stated in sec. 4.1.3, checking the consistency of the geometry of a scene is not trivial. In particular, it is very hard to do so, unless some assistance from the analyst is provided. If this effort from the analyst may be affordable when a single image is to be checked, it would be prohibitive to check geometric consistencies in video on a frame-by-frame basis. Existing works usually exploit phenomena connected to motion in order to detect editing. So far, two approaches have been proposed. One is based on artifacts introduced by video inpainting, that consists in hiding an object by automatically "filling the hole" with new synthetic pixels generated consistently with the (spatially) surrounding content. The other one is very specific, and it aims at revealing inconsistencies in the motion of objects in free-flight.

Doctoring detection based on ghost shadows

Zhang et al. [191] propose a method to detect video inpainting. Though originally developed for still images, this technique is also applicable to video. Nevertheless, frame-by-frame inpainting introduces annoying artifacts in the video, known as "ghost shadows", due to temporal discontinuity of the inpainted area. Authors begin by observing that these artifacts are well exposed in the Accumulative Difference Image (ADI), obtained by comparing the first frame image with every subsequent frame. However, ADI would also detect any moving object.
Therefore, the authors propose a method to automatically detect the presence of these artifacts, provided that the removed object was a moving object. The method estimates the moving foreground using two different approaches: standard moving foreground detection, performed using block matching, which produces a map of moving objects in the analyzed scene; and detection based on Accumulative Difference Image, which, if the video is original, should produce a similar map. If the two maps are not consistent, there is high probability that the analyzed video was doctored. The authors point out that only detection of forgery is possible, and no localization is provided. Experiments, performed on just a few real world video sequences, show that the method is robust against strong MPEG compression.

Doctoring detection based on inconsistencies in motion

If detecting geometrical inconsistencies in an inter-frame fashion is difficult, it is perhaps more difficult to detect physical inconsistencies. In fact, this would require to mix together tracking techniques and complex physical models to detect unexpected phenomena. Nevertheless, restricting the analysis to some specific scenarios, it is possible to develop ad-hoc techniques capable of such a task. This has been done, e.g., in the work by Conotter et al. [192]. An algorithm is proposed to detect physically implausible trajectories of objects in video sequences. The key idea is to explicitly model the three-dimensional trajectory of objects in free-flight (e.g. a ball flying towards the basket) and the corresponding two-dimensional projection into the image plane. The flying object is extracted from video, compensating camera motion if needed. The motion in the 3D space is estimated from 2D frames and compared to a plausible trajectory. If the detected deviation is large, the object is classified as tampered. Although being focused on a very specific scenario, the method inherits all the advantages that characterize forensics techniques based on physical and geometrical aspects. For example, performance does not depend on compression and video quality.

Copy detection

Wang and Farid [51] has proposed an approach that exploits multiple view geometry theory to understand if a video has been re-projected. This typically happens when someone records a movie from theater screen. The basic idea underlying this work is that, due to the angle of the camera relative to the screen, a perspective distortion is introduced in the acquired video, which shows up as a skew in the camera's intrinsic parameters. In [51] tools from multiple view geometry are adopted to estimate this kind of distortion. Since a minimum number of eight point correspondences between at least two views are required to estimate the camera skew, the authors propose to manually select a number of points (features) and to use standard tracking techniques, to maintain feature position for the entire scene. Results are good on synthetic examples (mean accuracy above 87%), and tests on a couple of real videos show that the technique works in practice.

4.3 Audio Editing

This section reviews several approaches to detect changes in audio material that have been applied subsequently to acquisition and, potentially, coding. In comparison to other media types, such as still images, relatively few results have been published so far.

Some approaches for edit detection are related to techniques used for the detection of acquisition or coding footprints. For instance, electric frequency network (ENF) information, which is used for acquisition-based footprints, can be also used to detect cuts or splicing edits in the audio material. On the other hand, specific footprints generated by audio compression methods might be used to detect subsequent editing of the audio material.

A major part of the techniques reviewed here stems from audio forensics research. General surveys of this field are provided, for instance, in [70, 193].

The techniques reviewed in this section can be classified into three groups. The first class aims at detecting cut-type edits, such as the removal of a part of an audio signal or the insertion of other audio data. The second group of techniques aims at detecting signal processing operations, such as the application of nonlinear effects or mixing of audio signals.

4.3.1 Cut-type edits

Cut-Type Edits refer to the removal of segments from an audio track, their rearrangement or concatenation. This section reviews a series of approaches that allow for their detection to identify altered audio material.

Time-domain detection methods

A time-domain method of detecting discontinuities in audio signals is proposed in [194].
Discrete differences $d^k x[n]$ are used as a primary tool:

$$d^1 x[n] = x[n] - x[n-1] \tag{4.1}$$

$$d^k x[n] = d^{k-1} x[n] - d^{k-1} x[n-1] \tag{4.2}$$

$$= \sum_{l=0}^{k} (-1)^l \binom{k}{l} x[n-l]\,. \tag{4.3}$$

As a discontinuity effectively is a jump or step function imposed on the audio signal, its first-order discrete difference forms a discrete impulse function, while the second-order difference forms an alternating, bipolar impulse.
The detection of discontinuities is performed as a two-step process. In a first step, a discrete difference operator of order k is applied to the signal $x[m]$, resulting in the signal $a[m]$. Basically, this operation represents a high-pass filter, passing rapid fluctuations in the signal (such as discontinuities) while attenuating the low-frequency content. In a second step, a sequence of correlation coefficients $r_{ab}[m]$ between $a[m]$ and the k-th order discrete difference of a discrete step function, $b[m]$ is calculated as

$$r_{ab}[m] = \sum_{l=0}^{k} b[m] a[m+l]\,. \tag{4.4}$$

The magnitude of $r_{ab}[m]$ is used to discriminate edit points. Note that the correlation coefficient differs from its conventional definition in two ways: First, subtracting the arguments by its mean is omitted, since the prior k-th order difference attenuates the low-frequency contents of the operands sufficiently. Second, normalization by the variance of the signals is not performed. Therefore, it appears sensible to incorporate the power of the audio signal into the determination of discontinuities. The length of the step function used to model the discontinuity as well as the order k of the difference operator influence the performance of discontinuity detection. Cooper states that a best detection can be provided with a second order difference and a step function of length 80.
Although the interpretation as a cross-correlation between the sequence of the differenced signal and a model of the discontinuity provides an intuitive interpretation of the search procedure, this algorithm can be also interpreted and implemented as a single FIR filtering operation. Apparently, the discrete difference operator of order k can be implemented as a finite convolution

$$a[m] = \sum_{l=0}^{k} x[m-l] h[l] = x[m] * h[m] \qquad \text{with } h[l] = (-1)^l \binom{k}{l}\,, \tag{4.5}$$

where $*$ denotes linear convolution. Since the model discontinuity $b[m]$ is formed by applying a k-th difference operator on a step function, it can be represented by the coefficients of a FIR filter. Likewise, the correlation (4.4) can be interpreted as a FIR filtering operation by reversing the order

of the coefficients. In this way, the correlation coefficient $r_{ab}[m]$ is determined by

$$r_{ab}[m] = \sum_{l=0}^{k} a[m]b'[m-l] \qquad \text{with } b'[l] = b[k-l] \tag{4.6}$$

$$= a[m] * h[m] * b'[m] \tag{4.7}$$

$$= a[m] * c[m] \qquad \text{with } c[m] = h[m] * b[m]\,, \tag{4.8}$$

since linear convolution is an associative operation. Thus, the detection of discontinuities proposed in [194] can be implemented as a FIR filtering operation. It appears worthwhile to evaluate the frequency response of the filter $c[m]$ and to investigate optimal filter responses for discriminating signal discontinuities.
Judging the suitability in the context of detecting tampered audio it must be noted that no attempt was made to discover edit concealment techniques, such as waveform alignment or cross-fading.

Cut detection and audio segment classification by fuzzy clustering

In [195], a technique to detect cuts in audio signals based on fuzzy clustering is proposed. Although the algorithm is described for MP3-encoded audio data, it can be used for arbitrary audio signals with only small modifications. Opposed to other approaches, this algorithm is also capable of detecting audio segments separated by fade-in, fade-out or crossfade operations.
In general, the algorithm operates on blocks (or frames) of audio data. In the proposed form, the frames of the decoded MP3 stream form these blocks. Cuts are only detected at block boundaries. An audio cut is detected by a change of the signal power of consecutive audio frames. Based on a fuzzy c-means clustering technique, a cut probability is assigned to each frame boundary.
In a second step, audio classification is applied to the audio segments between likely cut positions. This classification uses five classes: silence, music, speech, speech with music background, speech with noise background. For each segment, five features are extracted: average power, variance of power, average of center of gravity, variance of center of gravity and zero rate. Although some of these features are extracted directly from the MDCT (modified discrete cosine transform) representation of the MP3 signal, this algorithm can be straightforwardly adapted to other compressed formats or raw audio data. Based on this feature vector, classification is again performed by fuzzy c-means clustering.
However, the choice of c-means clustering is not the only possibility, since only binary classification is performed. For such tasks, other techniques, for instance Support Vector Machines [65], have also proven to be very effective.
The results of this classification are used to improve the quality of the cut detection algorithms. If two consecutive segments are classified identically, they are merged and the corresponding cut is discarded.
The algorithm is tested using a TV program consisting of different parts and shows reliable cut detection. However, the use of this approach for footprint detection is limited. First, the restriction to cuts between different classes of audio is not sensible for the considered application. Second, the assumption that a audio cut is accompanied by a change in signal power might not hold true for many envisaged forms of audio editing, especially in case of malicious tampering.

Scene change detection using eigen-audioframes

Another approach which also considers segmentation between different audiovisual scenes rather than detecting concealed cuts is proposed in [196]. They define an audio scene as a semantically consistent

segment, distinguishable by a dominant sound. A scene change occurs if the majority of dominant sources change. The audiovisual scene is considered to consist of a foreground signal, background audio, and transmission noise. It is assumed that at some point during a scene transition, only background audio and transmission noise are present. Thus, a scene transition exhibits relatively low levels of audio variance. It is therefore concluded that a scene change can be found as a large variation of the short-time energy in the background noise region.
To perform this detection, a method based on so-called eigen-audioframes in an audio subspace based on principal component analysis is derived. The eigen-audioframes are a set of K eigenvectors corresponding to the covariance of a matrix that is composed of the input audio signal arranged in non-overlapping frames. The determination of K and the selection of the eigenvalues corresponding to the eigen-audioframes is crucial to the overall performance of the algorithm. It turns out that the eigenvalues determining the audio frames should reside in the third quartile of the sequence of eigenvalue (sorted in descending order). As a consequence, the selected eigen-audioframes favor the detection of background noise and background audio.
Combining the eigen-audioframes into a matrix enables a transformation of audio frames into projected audio frames of dimension K. A difference measure $D(i)$ defined for each audio frame i is calculated as the Euclidean distance between the projection of the frame i and the projection of the sample average background noise. Frame indices where $D(i)$ assumes local minima are candidates for scene changes. To avoid false positives, an additional threshold that incorporates a-priori knowledge such as the minimum time between scene transitions is introduced.
The performance of the algorithm depends on the length of the audio frames. While longer frames reduce the probability of false positives, they also increase the number of missed scene transitions. Likewise, the frame length controls the detection of scene changes accompanied by audio fade-out/fade-in effects. To improve the detection performance of audiovisual scenes, the proposed algorithm is integrated with a method for detecting cuts in video scenes. This might offer new possibilities for detecting editing footprints in audiovisual media.

Techniques based on properties of lossy compression techniques

Signal features introduced by lossy audio compression techniques pose another possibility to detect cut-type alterations to an audio signal. Yang et. al. [197] propose a technique to detect cuts or edits in MP3 audio streams. Because it builds upon basic mechanisms of perceptual audio coding, it is likely to be applicable to other codecs.
The segmentation of an audio stream into frames of fixed size is the utilized first feature of MP3. These frames are transformed into the frequency domain using filter banks and a time-frequency transform such as the MDCT. The resulting coefficients are quantized prior to encoding. Since this quantization is controlled by a psychoacoustic model, many of the quantized MDCT coefficients equal zero. This property is utilized to detect frame offsets in audio signal, which indicate possible edits in the signal.
To detect such alterations, the decoded audio signal is filtered and transformed by a MDCT using different frame offsets. If the frame boundaries are identical to those in the original compression, then the MDCT coefficients show a significantly reduced number of nonzero coefficients, resulting from the prior quantization. For all other frame offsets, the number of zero-valued coefficients is significantly higher. By testing all possible frame offsets, it is possible to determine the offset used in the prior compression. If the audio signal is edited after decompression (deletion or insertion of samples), then the frame offset changes within the audio signal. Tests showed that the detection rates for deletions and insertions are consistently above 94% for different bitrates.
According to the description of the algorithm, the detection must be performed on the decompressed

signal. If the edited audio signal is encoded again, it is very likely that the subsequent quantization destroys the characteristic quantization pattern required to detect the frame offset. Thus, this paper falls somewhat short of its intended goal, the authentication of encoded MP3 audio.

Electric network frequency discontinuities

The use of ENF information has already been motivated in Section 2.3.2. However, the electric network frequency can also be used to detect cut-type edits. This approach is mentioned, amongst others, in [66, 198, 77].
In [199], two technique to detect and locate phase changes in the ENF signal are described. The first algorithm performs a sample rate reduction followed by filtering with a narrow band-pass filter whose passband region is centered around the nominal frequency of the electric network. In the filtered signal, phase discontinuities of the ENF signal manifest itself as temporary reductions of the signal's amplitude. Larger phase continuities result in larger amplitude reductions or, equivalently, amplitude modulation of the signal. This behavior is determined by the characteristics of the band-pass filter. Therefore, it would be interesting to investigate the influence of the filter design on the detection quality of the filter design.
No investigations are performed on the reasons for the perceived amplitude decrease. Apparently, this phenomenon follows directly from the spectral characteristics of a signal encountering a phase discontinuity. In a continuous, sinusoidal signal, the spectral energy is concentrated in a narrow band. Thus, a band-pass filter matched to this band yields a constantly high output. At a phase discontinuity, the energy of the signal is spread over a wide frequency band (which is typically audible as a click), leading to a temporary decrease of the signal energy in the passband region of the band-pass filter. Thus, the amplitude dip is only a secondary effect of a time-domain discontinuity in the ENF signal. Therefore, it would be worthwhile to investigate time-domain edit detection techniques such as [194] on the band-pass filtered portion of the input signal.
The second algorithm for ENF discontinuity detection proposed in [199] is based on a direct evaluation of the ENF phase. The signal is segmented into buffers and a DFT is performed on them. The authors state that the buffer duration must be an integral multiple of the period time of the nominal electric network frequency. Moreover, it is assumed that one frequency bin of the DFT matches this nominal frequency. By evaluating the phase of this frequency bin as a function of time, discontinuities in the ENF signal appear as abrupt phase changes. The authors state that this algorithm is robust with respect to the inherent alterations of the ENF. If the actual network frequency does not match a DFT frequency bin exactly, then the calculated phase becomes slowly time-varying. This behavior can be easily distinguished from the abrupt changes caused by ENF phase discontinuities. Although not explicitly stated by the authors, it is likely that this robustness also holds if the buffer duration differs slightly from an integer multiple of the ENF period size.
Both approaches are further expanded in [200].

Spectral distance measures

In the methods considered so far, edit points are generally found by detecting discontinuities in the audio signal. The detection of changes within the audio contents forms another possibility.
[199] proposes a method to detect edits in audio signals based on spectral distance measures. The method is particularly suited for discontinuities within periods of silence. Therefore, this method is useful for detecting malicious tampering attempts, because silent regions of the audio signal are likely candidates for editing.
Two different spectral measures are evaluated; namely a coefficient vector obtained from a linear predictor model and the real cepstral coefficient vector. These coefficient vectors are evaluated for

each input sample, and the Euclidean distance between subsequent vectors determines the spectral distance measure. To improve this detector, the algorithm is performed on the input signal both in forward and reverse time direction, and the maximum of both spectral differences is taken as final distance. However, no reasons nor objective performance evaluations are given to motivate this additional step.
The spectral distance detector is applied only to periods of silence, which are determined by a voice activity detector. This decision is motivated by large spectral distances that possibly occur in human speech, which might lead to false positives. On the other hand, it is argued that a large spectral distance measure in a period without vocal activity indicates an edit with high probability.
A related approach based on the the analysis of spectrograms is proposed in [77]. In this approach, the detection of edit locations is based on visual inspection. However, no criteria to implement this technique in an automated fashion are given.

4.3.2 Modifications by signal processing

Alterations by means of signal processing form another part of the editing history of an audio signal. Detection techniques typically aim at operations that occur at specific places in this history, including content production, the application of audio effects, or nonlinear processing that might be induced by digital forgeries.

Estimating mixing parameters

The parameters used in the production of audio contents, in particular in the processes of mixing and mastering, provide insight into the history of an audio track. In [201], a method to estimate these parameters is proposed. Both linear parameters (mixing gains or linear time-invariant filters) and nonlinear processes are targeted by this technique. The supported nonlinear elements are modeled as a gain controlled by a time-varying envelope that is calculated from a level measurement of the signal. This model includes a wide range of common audio effects such as limiters, compressors, or expanders.
The mixing parameters are determined by using a least-squares technique. A basic assumption of this approach is that the input tracks are linearly independent. It is shown that this property holds for most audio signals, even if source signals interfered during recording. While the determination of the mixing parameters is not unique in many cases, i.e., if two gains in a processing chain can be adjusted to yield the same final gain, it is shown that the least-squares algorithm achieves a non-degenerate solution in such cases. While linear parameters such as gains or filter coefficients can be calculated for the complete audio signal, the gain envelope corresponding to a nonlinear processing element is calculated frame-wise. For each audio frame, this envelope is modeled by a polynomial whose coefficients are determined using a least-squares criterion.
However, the proposed parameter estimation requires the input tracks of the mixing process to be present. This is a major drawback for the use in footprint detection, which is typically considered as a blind process. In addition, the size of the least-squares problem grows with the number of estimated parameters. As an example, if filters are modeled in the mixing process, the size of involved matrices grows linear with the assumed filter order. Besides the increased computational complexity, such problems are prone to ill-conditioning. A high sensitivity to noise is noted by the authors as a main disadvantage of the algorithm. It appears likely that this sensitivity is related to the conditioning of the problem.
A related approach is proposed in [202]. The primary intention of this paper is to perform automatic mixing of multichannel recordings. A set of mixing gains is adapted such that the distance of a spec-

tral histogram feature of the generated mix to a predefined target spectral histogram is minimized. This adaptation is implemented by a genetic algorithm. To use this algorithm to estimate mixing parameters, the spectral histogram of the final audio signal is used as target.
Compared to [201], this approach has two main drawbacks. First, the estimation is limited to a set of mixing gains. Second, the algorithms used require a high computational effort. Like the preceding approach it requires the original tracks of the recording, it is not directly for blind footprint detection techniques.

Detecting nonlinear processing using bispectral analysis

The detection of nonlinear processing might reveal subsequent modification of an audio signal and provides information about the history of audio data. In [203], a technique to identify nonlinear processing applied to human speech signals to detect digital forgeries is proposed.
The key assumption for this approach is that natural audio signals as the human voice have weak higher-order statistical correlations, documented by a large set of speaker data. Nonlinear processing introduces new harmonics with correlated phases into the signal. For this reason, higher-order correlations are significantly higher if such edits have been applied. The bispectrum or its normalized representation, the bicoherence, are used as measures to detect such higher-order correlations.
The utility of these measures is shown using different types of edits. For a static nonlinearity applied to the whole duration of a speech sequence, the magnitude and the phase offset of the bicoherence increase significantly. At the same time, standard statistical features such as the power spectrum do not show characteristic changes. In a second test, this nonlinearity is applied only to a small region of the audio signal. It is shown that bispectral analysis is able to detect and locate this modification reliably. As a third scenario, cut-type edits between artifical test signals are investigated. Although a sophisticated technique (a Laplacian pyramid) is used to yield a seamless waveform without audible discontinuities, the bispectral measures show a significant increase at the region of alteration. A fourth test evaluates the detection of cut-type edits on human speech, where passages are deleted from a sentence, changing its meaning. In this scenario, the bispectral activity shows a significant increase at the location of the edits, too.
Although this approach seems suitable for recordings of natural speech, it appears necessary to investigate this approach for other audio contents. At the one hand, the higher-order statistics of other types of audio signals must be evaluated. There are several classes of signals that do not fulfill this statistical properties, for instance synthesized sound or instruments that are based on nonlinear effects, such as electric guitars.

Detecting multiple effects in guitar recordings

An approach to detect combinations of multiple audio effects in electric guitar recordings is described in [204]. This technique operates on monophonic guitar recordings and is based on semantic music analysis techniques. Arbitrary combinations of six audio effects commonly used in guitar recordings, classified into three groups, are detected from the sustain part of guitar tones. The proposed algorithm consists of several steps and follows the general framework of systems for semantic music analysis, e.g. [205]. A preprocessing step extracts the sustain part from guitar tones. After segmentation into overlapping frames, the signal is transformed into the frequency domain. Feature extraction computes an extensive vector of 541 features for each frame. These features fall into three categories (spectral, cepstral and harmonic features). Due to the nature of the audio effects considered, special emphasis is placed on harmonic features. To reduce the dimension of the feature vectors, two feature reduction techniques, namely Inertia Ratio Maximization (IRM) [206] and Linear Discriminant Analysis (LDA) [207] are applied. The detection of multiple, cascaded

effects represents a multi-label classification task. This multi-label classification is transformed into a single-label classification problem by considering each possible combination of audio effects as a separate class. Consequently, the database used for classification consists of a multitude of guitar tones, each processed with each possible combination of the audio effects. After training, the classification accuracy generally exceeds 95%, the majority of false classifications is due to incomplete detection of multiple effects.

Beyond the proposed application, this approach also shows a number of interesting features for other types of signals or the detection of other effects. First, the detection and selection of a set of signal components suitable for a given classificator appears a sensible approach. Second, the framework of semantic music analysis, using feature extraction, feature reduction and classification based on machine learning techniques offers new possibilities for detecting characteristic footprints of common processing techniques.

Bibliography

[1] A. Swaminathan, M. Wu, and K. J. R. Liu, "A pattern classification framework for theoretical analysis of component forensics," in *ICASSP*. IEEE, 2008, pp. 1665–1668.

[2] C. McKay, A. Swaminathan, H. Gou, and M. Wu, "Image acquisition forensics: Forensic analysis to identify imaging source," in *ICASSP*. IEEE, 2008, pp. 1657–1660.

[3] O. Çeliktutan, B. Sankur, and I. Avcibas, "Blind identification of source cell-phone model," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 553–566, 2008.

[4] C.-T. Li and Y. Li, "Digital camera identification using colour-decoupled photo response non-uniformity noise pattern," in *ISCAS*. IEEE, 2010, pp. 3052–3055.

[5] B. Mahdian and S. Saic, "A bibliography on blind methods for identifying image forgery," *Sig. Proc.: Image Comm.*, vol. 25, no. 6, pp. 389–399, 2010.

[6] M. Chen, J. J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.

[7] Y. Sutcu, S. Bayram, H. T. Sencar, and N. D. Memon, "Improvements on sensor noise based source camera identification," in *ICME*. IEEE, 2007, pp. 24–27.

[8] B.-B. Liu, Y. Hu, and H.-K. Lee, "Source camera identification from significant noise residual regions," in *ICIP*. IEEE, 2010, pp. 1749–1752.

[9] C.-T. Li, "Source camera identification using enahnced sensor pattern noise," in *ICIP*. IEEE, 2009, pp. 1509–1512.

[10] T. Filler, J. J. Fridrich, and M. Goljan, "Using sensor pattern noise for camera model identification," in *ICIP*. IEEE, 2008, pp. 1296–1299.

[11] M. Goljan, J. J. Fridrich, and T. Filler, "Managing a large database of camera fingerprints," in *Media Forensics and Security*, ser. SPIE Proceedings, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7541. SPIE, 2010, p. 754108.

[12] G. J. Bloy, "Blind camera fingerprinting and image clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 3, pp. 532–534, 2008.

[13] K. Rosenfeld and H. T. Sencar, "A study of the robustness of prnu-based camera identification," in *Media Forensics and Security*, ser. SPIE Proceedings, E. J. Delp, J. Dittmann, N. D. Memon, and P. W. Wong, Eds., vol. 7254. SPIE, 2009, p. 72540.

[14] A. Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," *Signal Processing, IEEE Transactions on*, vol. 53, no. 10, pp. 3948 – 3959, oct. 2005.

[15] S. Bayram, H. T. Sencar, N. D. Memon, and I. Avcibas, "Source camera identification based on cfa interpolation," in *ICIP (3)*, 2005, pp. 69–72.

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

[17] H. Cao and A. C. Kot, "Accurate detection of demosaicing regularity from output images," in *ISCAS*. IEEE, 2009, pp. 497–500.

[18] M. Kirchner, "Efficient estimation of cfa pattern configuration in digital camera images," in *Media Forensics and Security*, ser. SPIE Proceedings, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7541. SPIE, 2010, p. 754111.

[19] M. Kirchner and R. Böhme, "Synthesis of color filter array pattern in digital images," in *Media Forensics and Security*, ser. SPIE Proceedings, E. J. Delp, J. Dittmann, N. D. Memon, and P. W. Wong, Eds., vol. 7254. SPIE, 2009, p. 72540.

[20] A. Gallagher and T. Chen, "Image authentication by detecting traces of demosaicing," in *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on*, june 2008, pp. 1 –8.

[21] T. V. Lanh, S. Emmanuel, and M. S. Kankanhalli, "Identifying source cell phone using chromatic aberration," in *ICME*. IEEE, 2007, pp. 883–886.

[22] K. S. Choi, E. Y. Lam, and K. K. Y. Wong, "Automatic source camera identification using the intrinsic lens radial distortion," *Opt. Express*, vol. 14, no. 24, pp. 11 551–11 565, Nov 2006. [Online]. Available: http://www.opticsexpress.org/abstract.cfm?URI=oe-14-24-11551

[23] A. E. Dirik, H. T. Sencar, and N. D. Memon, "Digital single lens reflex camera identification from traces of sensor dust," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 539–552, 2008.

[24] C. Riess and E. Angelopoulou, "Scene illumination as an indicator of image manipulation," in *Information Hiding*, 2010, pp. 66–80.

[25] H. Farid and J. Kosecká, "Estimating planar surface orientation using bispectral analysis," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2154–2160, 2007.

[26] M. K. Johnson and H. Farid, "Exposing digital forgeries in complex lighting environments," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3-1, pp. 450–461, 2007.

[27] ——, "Exposing digital forgeries through specular highlights on the eye," in *Information Hiding*, ser. Lecture Notes in Computer Science, T. Furon, F. Cayre, G. J. Doërr, and P. Bas, Eds., vol. 4567. Springer, 2007, pp. 311–325.

[28] S. Lin and L. Zhang, "Determining the radiometric response function from a single grayscale image," in *CVPR (2)*. IEEE Computer Society, 2005, pp. 66–73.

[29] T.-T. Ng, S.-F. Chang, and M.-P. Tsui, "Using geometry invariants for camera response function estimation," in *CVPR*. IEEE Computer Society, 2007.

[30] 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 *ACM Multimedia*, H. Zhang, T.-S. Chua, R. Steinmetz, M. S. Kankanhalli, and L. Wilcox, Eds. ACM, 2005, pp. 239–248.

[31] H. Cao and A. C. Kot, "Identification of recaptured photographs on lcd screens," in *ICASSP*. IEEE, 2010, pp. 1790–1793.

[32] H. Yu, T.-T. Ng, and Q. Sun, "Recaptured photo detection using specularity distribution," in *ICIP*. IEEE, 2008, pp. 3140–3143.

[33] X. Gao, T.-T. Ng, B. Qiu, and S.-F. Chang, "Single-view recaptured image detection based on physics-based features," in *ICME*. IEEE, 2010, pp. 1469–1474.

[34] M. Goljan, J. Fridrich, and y. . . o. . A. t. . . Jan Luk, title = Camera Identification from Printed Images.

[35] X. Gao, B. Qiu, J. Shen, T.-T. Ng, and Y. Q. Shi, "A smart phone image database for single image recapture detection," in *IWDW*, ser. Lecture Notes in Computer Science, H.-J. Kim, Y. Q. Shi, and M. Barni, Eds., vol. 6526. Springer, 2010, pp. 90–104.

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

[37] C.-H. Choi, M.-J. Lee, and H.-K. Lee, "Scanner identification using spectral noise in the frequency domain," in *ICIP*. IEEE, 2010, pp. 2121–2124.

[38] N. Khanna and E. J. Delp, "Intrinsic signatures for scanned documents forensics : Effect of font shape and size," in *ISCAS*. IEEE, 2010, pp. 3060–3063.

[39] A. E. Dirik, H. T. Sencar, and N. D. Memon, "Flatbed scanner identification based on dust and scratches over scanner platen," in *ICASSP*. IEEE, 2009, pp. 1385–1388.

[40] N. Khanna, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," in *ICASSP*. IEEE, 2008, pp. 1653–1656.

[41] R. Caldelli, I. Amerini, and F. Picchioni, "A dft-based analysis to discern between camera and scanned images," *IJDCF*, vol. 2, no. 1, pp. 21–29, 2010.

[42] S. Dehnie, H. T. Sencar, and N. D. Memon, "Digital image forensics for identifying computer generated and digital camera images," in *ICIP*. IEEE, 2006, pp. 2313–2316.

[43] W. Li, T. Zhang, E. Zheng, and X. Ping, "Identifying photorealistic computer graphics using second-order difference statistics," in *FSKD*, M. Li, Q. Liang, L. Wang, and Y. Song, Eds. IEEE, 2010, pp. 2316–2319.

[44] A. E. Dirik, S. Bayram, H. T. Sencar, and N. D. Memon, "New features to identify computer generated images," in *ICIP (4)*. IEEE, 2007, pp. 433–436.

[45] F. Pan and J. Huang, "Discriminating computer graphics images and natural images using hidden markov tree model," in *IWDW*, ser. Lecture Notes in Computer Science, H.-J. Kim, Y. Q. Shi, and M. Barni, Eds., vol. 6526. Springer, 2010, pp. 23–28.

[46] W. van Houten, Z. J. M. H. Geradts, K. Franke, and C. J. Veenman, "Verification of video source camera competition (camcom 2010)," in *ICPR Contests*, ser. Lecture Notes in Computer Science, D. Ünay, Z. Çataltepe, and S. Aksoy, Eds., vol. 6388. Springer, 2010, pp. 22–28.

[47] W. van Houten and Z. J. M. H. Geradts, "Source video camera identification for multiply compressed videos originating from youtube," *Digital Investigation*, vol. 6, no. 1-2, pp. 48–60, 2009.

[48] ——, "Using sensor noise to identify low resolution compressed videos from youtube," in *IWCF*, ser. Lecture Notes in Computer Science, Z. J. M. H. Geradts, K. Franke, and C. J. Veenman, Eds., vol. 5718. Springer, 2009, pp. 104–115.

[49] W. Wang, S. Member, and H. Farid, "Exposing digital forgeries in interlaced and de-interlaced video," *IEEE Transactions on Information Forensics and Security*, vol. 2007, 2007.

[50] J.-W. Lee, M.-J. Lee, T.-W. Oh, S.-J. Ryu, and H.-K. Lee, "Screenshot identification using combing artifact from interlaced video," in *Proceedings of the 12th ACM workshop on Multimedia and security*, ser. MM&Sec '10. New York, NY, USA: ACM, 2010, pp. 49–54. [Online]. Available: http://doi.acm.org/10.1145/1854229.1854240

[51] W. Wang and H. Farid, "Detecting re-projected video," in *Information Hiding*, ser. Lecture Notes in Computer Science, K. Solanki, K. Sullivan, and U. Madhow, Eds., vol. 5284. Springer, 2008, pp. 72–86.

[52] M. Kobayashi, T. Okabe, and Y. Sato, "Detecting forgery from static-scene video based on inconsistency in noise level functions," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 883–892, 2010.

[53] ——, "Detecting video forgeries based on noise characteristics," in *PSIVT*, ser. Lecture Notes in Computer Science, T. Wada, F. Huang, and S. Lin, Eds., vol. 5414. Springer, 2009, pp. 306–317.

[54] M.-J. Lee, K.-S. Kim, and H.-K. Lee, "Digital cinema watermarking for estimating the position of the pirate," *IEEE Transactions on Multimedia*, vol. 12, no. 7, pp. 605–621, 2010.

[55] M.-J. Lee, K.-S. Kim, H.-Y. Lee, T.-W. Oh, Y.-H. Suh, and H.-K. Lee, "Robust watermark detection against d-a/a-d conversion for digital cinema using local auto-correlation function," in *ICIP*. IEEE, 2008, pp. 425–428.

[56] F. Chang and H. Huang, "Electrical network frequency as a tool for audio concealment process," in *Intelligent Information Hiding and Multimedia Signal Processing, International Conference on*, vol. 0. Los Alamitos, CA, USA: IEEE Computer Society, 2010, pp. 175–178.

[57] D. Boss, "Visualization of magnetic features on analogue audiotapes is still an important task," in *Audio Engineering Society Conference: 39th International Conference: Audio Forensics: Practices and Challenges*, 6 2010. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=15499

[58] T. Owen, "Forensic audio and video-theory and application," *Journal of the Audio Engineering Society*, vol. 36, pp. 34–41, 1988.

[59] C. Kraetzer, K. Qian, M. Schott, and J. Dittmann, "A context model for microphone forensics and its application in evaluations," in *SPIE Conference on Media Watermarking, Security, and Forensics*, 2011.

[60] S. Molau, M. Pitz, R. Schluter, and H. Ney, "Computing Mel-frequency cepstral coefficients on the power spectrum," in *IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP)*, vol. 1. Citeseer, 2001, pp. 1–4. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.23.6535\&rep=rep1\&type=pdf

[61] R. Buchholz, C. Kraetzer, and J. Dittmann, "Microphone classification using fourier coefficients," in *Proceedings of the International Workshop on Information Hiding*. Darmstadt, Germany: Springer, 2009, pp. 235–246. [Online]. Available: http://www.springerlink.com/index/2278721855474578.pdf

[62] C. Kraetzer, M. Schott, and J. Dittmann, "Unweighted fusion in microphone forensics using a decision tree and linear logistic regression models," in *Proceedings of the 11th ACM workshop on Multimedia and security*, ser. MM&Sec '09. New York, NY, USA: ACM, 2009, pp. 49–56. [Online]. Available: http://doi.acm.org/10.1145/1597817.1597827

[63] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: a first practical evaluation on microphone and environment classification," in *Proceedings of the 9th workshop on Multimedia & security*, ser. MM&Sec '07. New York, NY, USA: ACM, 2007, pp. 63–74. [Online]. Available: http://doi.acm.org/10.1145/1288869.1288879

[64] D. Garcia-Romero and C. Y. Espy-Wilson, "Automatic acquisition device identification from speech recordings," in *ICASSP*. IEEE, 2010, pp. 1806–1809.

[65] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett, "New support vector algorithms," *Neural Computation*, vol. 12, no. 5, pp. 1207–1245, 2000. [Online]. Available: http://www.mitpressjournals.org/doi/abs/10.1162/089976600300015565

[66] M. Kajstura, A. Trawinska, and J. Hebenstreit, "Application of the electrical network frequency (ENF) criterion: A case of a digital recording," *Forensic Science International*, vol. 155, no. 2-3, pp. 165 – 171, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0379073804007662

[67] C. Grigoras, "Digital audio recording analysis: the electric network frequency (enf) criterion," *International Journal of Speech Language and the Law*, vol. 12, no. 1, pp. 1350–1771, 2005. [Online]. Available: http://www.equinoxjournals.com/IJSLL/article/view/525

[68] ——, "Applications of enf criterion in forensic audio, video, computer and telecommunication analysis," *Forensic Science International*, vol. 167, no. 2-3, pp. 136 – 145, 2007, selected Articles of the 4th European Academy of Forensic Science Conference (EAFS2006) June 13-16, 2006 Helsinki, Finland. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0379073806004312

[69] ——, "Applications of enf analysis method in forensic authentication of digital audio and video recordings," in *Audio Engineering Society Convention 123*, 10 2007. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14331

[70] E. B. Brixen, "Techniques for the authentication of digital audio recordings," in *Audio Engineering Society Convention 122*, 5 2007. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=13999

[71] ——, "Further investigation into the ENF criterion for forensic authentication," in *Audio Engineering Society Convention 123*, 10 2007. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14333

[72] A. J. Cooper, "The electric network frequency (ENF) as an aid to authenticating forensic digital audio recordings an automated approach," in *Audio Engineering Society Conference: 33rd International Conference: Audio Forensics-Theory and Practice*, 6 2008. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14411

[73] R. W. Sanders, "Digital audio authenticity using the electric network frequency," in *Audio Engineering Society Conference: 33rd International Conference: Audio Forensics-Theory and Practice*, 6 2008. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14403

[74] R. W. Sanders and P. S. Popolo, "Extraction of electric network frequency signals from recordings made in a controlled magnetic field," in *Audio Engineering Society Convention 125*, 10 2008. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14794

[75] C. Grigoras, "Applications of enf analysis in forensic authentication of digital audio and video recordings," *Journal of the Audio Engineering Society*, vol. 57, no. 9, pp. 643–661, Sep. 2009.

[76] E. B. Brixen, "ENF; quantification of the magnetic field," in *Audio Engineering Society Conference: 33rd International Conference: Audio Forensics-Theory and Practice*, 6 2008. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14412

[77] R. Korycki, "Methods of time-frequency analysis in authentication of digital audio recordings," *International Journal of Electronics and Telecommunications*, vol. 56, no. 3, pp. 257–262, Sep. 2010. [Online]. Available: http://versita.metapress.com/openurl.asp?genre=article&id=doi:10.2478/v10177-010-0033-0

[78] L. Cohen, *Time-Frequency Analysis*. Englewood Cliffs, NJ: Prentice Hall PTR, 1995.

[79] M. Huijbregtse and Z. Geradts, "Using the ENF criterion for determining the time of recording of short digital audio recordings," in *Computational Forensics*, ser. Lecture Notes in Computer Science, Z. Geradts, K. Franke, and C. Veenman, Eds. Springer Berlin / Heidelberg, 2009, vol. 5718, pp. 116–124. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03521-0_11

[80] R. Korycki, "Methods of Time-Frequency analysis in authentication of digital audio recordings," *International Journal of Electronics and Telecommunications*, vol. 56, no. 3, pp. 257–262, 2010. [Online]. Available: http://dx.doi.org/10.2478/v10177-010-0033-0

[81] R. Maher, "Modeling and signal processing of acoustic gunshot recordings," in *12th Digital Signal Processing Workshop, and 4th Signal Processing Education Workshop*, Sep. 2006, pp. 257–261.

[82] R. C. Maher and S. R. Shaw, "Directional aspects of forensic gunshot recordings," in *Proceedings of the 39th International AES Conference Audio Forensics: Practices and Challenges*, Hillerød, Denmark, Jun. 2010, pp. 127–132.

[83] H. Malik and H. Farid, "Audio forensics from acoustic reverberation," in *Proveedings of the IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, no. iid, 2010, pp. 1710–1713. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5495479

[84] R. Ratnam, D. L. Jones, B. C. Wheeler, C. R. OBrien Jr., William D.and Lansing, and A. S. Feng, "Blind estimation of reverberation time," *Journal of the Acoustical Society of America*, vol. 114, no. 5, pp. 2877–2892, Nov. 2003. [Online]. Available: http://dx.doi.org/10.1121/1.1616578

[85] H. W. Löllmann and P. Vary, "Estimation of the reverberation time in noisy environments," in *Proceedings of the 11th International Workshop on Acoustic Echo and Noise Control*, Seattle, WA, Sep. 2008.

[86] J. R. Hopgood and P. J. Rayner, "Blind single channel deconvolution using nonstationary signal processing," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 5, pp. 476 – 488, Sep. 2003.

[87] E. A. P. Habets, "Single- and multi-microphone speech dereverberation using spectral enhancement," Ph.D. dissertation, Technische Universiteit Eindhoven, Eindhoven, The Netherlands, Jun. 2007.

[88] T. Yoshioka, T. Nakatani, and M. Miyoshi, "An integrated method for blind source separation and dereverberation of convolutive audio mixture," in *Proceedings of the 16th European Signal Processing Conference (EUSIPCO 2008)*, Lausanne, Switzerland, Aug. 2008.

[89] K. Furuya and A. Kataoka, "Robust speech dereverberation using multichannel blind deconvolution with spectral subtraction," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1579–1591, Jul. 2007.

[90] T. Nakatani, M. Miyoshi, and K. Kinoshita, "One microphone blind dereverberation based on quasi-periodicity of speech signals," in *Advances in Neural Information Processing Systems 16*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, vol. 16. [Online]. Available: http://books.nips.cc/papers/files/nips16/NIPS2003_SP06.pdf

[91] M. Triki and D. T. Slock, "Blind deverberation of a single source based on multichannel linear prediction," in *Proceedings of the 2005 International Workshop on Acoustic Echo and Noise Control*, 2005.

[92] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Blind speech dereverberation with multi-channel linear prediction based on short time Fourier transform representation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008)*, Apr. 2008, pp. 85–88.

[93] N. Yasuraoka, T. Yoshioka, T. Nakatani, A. Nakamura, and H. Okuno, "Music dereverberation using harmonic structure source model and Wiener filter," in *2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, Mar. 2010, pp. 53–56.

[94] W. Wang, J. Dong, and T. Tan, "A survey of passive image tampering detection," in *IWDW*, ser. Lecture Notes in Computer Science, A. T. S. Ho, Y. Q. Shi, H. J. Kim, and M. Barni, Eds., vol. 5703. Springer, 2009, pp. 308–322.

[95] G. Wallace, "The jpeg still picture compression standard," *Consumer Electronics, IEEE Transactions on*, vol. 38, no. 1, pp. xviii –xxxiv, feb 1992.

[96] Z. Fan and R. L. de Queiroz, "Maximum likelihood estimation of JPEG quantization table in the identification of bitmap compression history," in *ICIP*, 2000.

[97] ——, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 230–235, 2003.

[98] W. Luo, J. Huang, and G. Qiu, "JPEG error analysis and its applications to digital image forensics," *Information Forensics and Security, IEEE Transactions on*, vol. 5, no. 3, pp. 480 –491, sept. 2010.

[99] W. S. Lin, S. K. Tjoa, H. V. Zhao, and K. J. R. Liu, "Digital image source coder forensics via intrinsic fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 460–475, 2009.

[100] S. Ye, Q. Sun, and E.-C. Chang, "Detecting digital image forgeries by measuring inconsistencies of blocking artifact," in *ICME*. IEEE, 2007, pp. 12–15.

[101] R. Neelamani, R. L. de Queiroz, Z. Fan, S. Dash, and R. G. Baraniuk, "JPEG compression history estimation for color images," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1365–1378, 2006.

[102] T. Pevny and J. Fridrich, "Estimation of primary quantization matrix for steganalysis of double-compressed JPEG images," *Proceedings of SPIE*, vol. 6819, pp. 681 911–681 911–13, 2008. [Online]. Available: http://link.aip.org/link/PSISDG/v6819/i1/p681911/s1&Agg=doi

[103] J. Lukás and J. Fridrich, "Estimation of primary quantization matrix in double compressed jpeg images," in *Proc. of DFRWS*, 2003.

[104] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double MPEG compression," in *MM&Sec*, S. Voloshynovskiy, J. Dittmann, and J. J. Fridrich, Eds. ACM, 2006, pp. 37–47.

[105] J. He, Z. Lin, L. Wang, and X. Tang, "Detecting doctored JPEG images via DCT coefficient analysis," in *Lecture Notes in Computer Science*. Springer, 2006, pp. 423–435.

[106] J. Fridrich, M. Goljan, and D. Hogea, "Attacking the outguess," in *Proceedings of the 3rd Information Hiding Workshop on Multimedia and Security*, 2002.

[107] M. Xian-zhe, N. Shao-zhang, and Z. Jian-chen, "Tamper detection for shifted double jpeg compression," in *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, oct. 2010, pp. 434 –437.

[108] D. Fu, Y. Q. Shi, and W. Su, "A generalized benfords law for jpeg coefficients and its applications in image forensics," in *Proceedings of SPIE, Volume 6505, Security, Steganography and Watermarking of Multimedia Contents IX*, vol. 6505, Jan. 28 – Feb. 1, 2009, pp. 39–48.

[109] F. Benford, "The law of anomalous numbers," *Proceedings of the American Philosophical Society*, vol. 78, no. 4, pp. 551–572, 1938.

[110] B. Li, Y. Q. Shi, and J. Huang, "Detecting doubly compressed JPEG images by using mode based first digit features," in *MMSP*. IEEE Signal Processing Society, 2008, pp. 730–735.

[111] M. Barni, A. Costanzo, and L. Sabatini, "Identification of cut & paste tampering by means of double-JPEG detection and image segmentation," in *ISCAS*. IEEE, 2010, pp. 1687–1690.

[112] Z. Qu, W. Luo, and J. Huang, "A convolutive mixing model for shifted double JPEG compression with application to passive image authentication," in *ICASSP*. IEEE, 2008, pp. 1661–1664.

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

[114] Z. Wang, A. C. Bovik, and B. L. Evans, "Blind measurement of blocking artifacts in images," in *ICIP*, 2000.

[115] H. Liu and I. Heynderickx, "A no-reference perceptual blockiness metric," in *ICASSP*. IEEE, 2008, pp. 865–868.

[116] S. K. Tjoa, W.-Y. S. Lin, H. V. Zhao, and K. J. R. Liu, "Block size forensic analysis in digital images," in *ICASSP (1)*. IEEE, 2007, pp. 633–636.

[117] Y.-L. Chen and C.-T. Hsu, "Image tampering detection by blocking periodicity analysis in JPEG compressed images," in *MMSP*. IEEE Signal Processing Society, 2008, pp. 803–808.

[118] J. Zhang, H. Wang, and Y. Su, "Detection of double-compression in jpeg2000 images," in *Intelligent Information Technology Application, 2008. IITA '08. Second International Symposium on*, vol. 1, dec. 2008, pp. 418 –421.

[119] Z. Fan, S. Wang, S. Li, and Y. Zhang, "Detection of double-compression in jpeg2000 by using markov features," in *Computing and Intelligent Systems*, ser. Communications in Computer and Information Science, Y. Wu, Ed. Springer Berlin Heidelberg, 2011, vol. 234, pp. 441–449.

[120] M. C. Stamm, S. K. Tjoa, W. S. Lin, and K. J. R. Liu, "Anti-forensics of JPEG compression," in *ICASSP*. IEEE, 2010, pp. 1694–1697.

[121] ——, "Undetectable image tampering through JPEG compression anti-forensics," in *ICIP*. IEEE, 2010, pp. 2109–2112.

[122] M. C. Stamm and K. J. R. Liu, "Wavelet-based image compression anti-forensics," in *ICIP*. IEEE, 2010, pp. 1737–1740.

[123] Y. Chen, K. S. Challapali, and M. Balakrishnan, "Extracting coding parameters from precoded MPEG-2 video," in *ICIP (2)*, 1998, pp. 360–364.

[124] M. Tagliasacchi and S. Tubaro, "Blind estimation of the QP parameter in H.264/AVC decoded video," in *Image Analysis for Multimedia Interactive Services (WIAMIS), 2010 11th International Workshop on*, april 2010, pp. 1 –4.

[125] G. Valenzise, M. Tagliasacchi, and S. Tubaro, "Estimating QP and motion vectors in H.264/AVC video from decoded pixels," in *Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence*, ser. MiFor '10. New York, NY, USA: ACM, 2010, pp. 89–92. [Online]. Available: http://doi.acm.org/10.1145/1877972.1877995

[126] H. Li and S. Forchhammer, "MPEG2 video parameter and no reference PSNR estimation," in *Picture Coding Symposium, 2009. PCS 2009*, may 2009, pp. 1 –4.

[127] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, "A method of estimating coding PSNR using quantized DCT coefficients," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 16, no. 2, pp. 251–259, 2006.

[128] T. Brandão and T. R. M. P. Queluz, "No-reference quality assessment of H.264/AVC encoded video," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 20, no. 11, pp. 1437–1447, 2010.

[129] A. R. Reibman and D. Poole, "Characterizing packet-loss impairments in compressed video," in *ICIP (5)*. IEEE, 2007, pp. 77–80.

[130] A. R. Reibman, V. A. Vaishampayan, and Y. Sermadevi, "Quality monitoring of video over a packet network," *IEEE Transactions on Multimedia*, vol. 6, no. 2, pp. 327–334, 2004.

[131] M. Naccari, M. Tagliasacchi, and S. Tubaro, "No-reference video quality monitoring for H.264/AVC coded video," *IEEE Transactions on Multimedia*, vol. 11, no. 5, pp. 932–946, 2009.

[132] G. Valenzise, S. Magni, M. Tagliasacchi, and S. Tubaro, "Estimating channel-induced distortion in H.264/AVC video without bitstream information," in *Quality of Multimedia Experience (QoMEX), 2010 Second International Workshop on*, june 2010, pp. 100 –105.

[133] W. Luo, M. Wu, and J. Huang, "MPEG recompression detection based on block artifacts," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, ser. Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, vol. 6819, Mar. 2008.

[134] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double quantization," in *Proceedings of the 11th ACM workshop on Multimedia and security*, ser. MM&Sec '09. New York, NY, USA: ACM, 2009, pp. 39–48. [Online]. Available: http://doi.acm.org/10.1145/1597817.1597826

[135] ISO/IEC 11172-3:1993, "Information technology – Coding of moving pictures and associated audio for digital storagemedia at up to about 1.5 Mbit/s – Part 3: Audio," The Moving Picture Experts Group (MPEG). [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=22412

[136] K. Brandenburg, "MP3 and AAC explained," in *Proceedings of the AES 17th International Conference on High Quality Audio Coding*. Florence, Italy: Audio Engineering Society, 1999. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=8079

[137] J. Princen, A. Johnson, and A. Bradley, "Subband/transform coding using filter bank designs based on time domain aliasing cancellation," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 12, apr 1987, pp. 2161–2164.

[138] J. H. Rothweiler, "Polyphase quadrature filters – A new subband coding technique," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 8, apr 1983, pp. 1280–1283.

[139] B. Edler, "Codierung von Audiosignalen mit überlappender Transformation und adaptiven Fensterfunktionen," in *Kleinheubacher Tagung*, Oct. 1989, [in german].

[140] F. Baumgarte, C. Ferekidis, and H. Fuchs, "A nonlinear psychoacoustic model applied to ISO/MPEG Layer 3 coder," in *Proceedings of the 99th Convention of Audio Engineering Society*, 10 1995. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=7679

[141] S. Moehrs, J. Herre, and R. Geiger, "Analysing decompressed audio with the "Inverse Decoder" – towards an operative algorithm," in *Proceedings of the 112th Convention of Audio Engineering Society*. Munich, Germany: Audio Engineering Society, 2002, pp. 1–22. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=11346

[142] B. D'Alessandro and Y. Q. Shi, "Mp3 bit rate quality detection through frequency spectrum analysis," in *Proceedings of the 11th ACM workshop on Multimedia and security (MM&Sec)*. New York, New York, USA: ACM Press, 2009, pp. 57–61. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1597817.1597828

[143] J. Herre and M. Schug, "Analysis of decompressed audio - the "Inverse Decoder"," in *Proceedings of the 109th Convention of Audio Egineering Society*, Los Angeles, California, USA, 2000.

[144] R. Yang, Y.-Q. Shi, and J. Huang, "Defeating fake-quality mp3." New York, New York, USA: ACM Press, 2009, pp. 117–124. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1597817.1597838

[145] A. J. Fridrich, B. D. Soukal, and b. . i. y. . . A Jan Luk, title = Detection of copy-move forgery in digital images.

[146] W. Q. Luo, J. W. Huang, and G. P. Qiu, "Robust detection of region-duplication forgery in digital image," in *ICPR*, 2006, pp. IV: 746–749. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/ICPR.2006.1003

[147] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Dartmouth College, Computer Science, Hanover, NH, Tech. Rep. TR2004-515, Aug. 2004. [Online]. Available: http://www.cs.dartmouth.edu/reports/TR2004-515.pdf

[148] S. Bayram, H. T. Sencar, and N. D. Memon, "An efficient and robust method for detecting copy-move forgery," in *ICASSP*. IEEE, 2009, pp. 1053–1056.

[149] Q. Wu, S. Wang, and X. Zhang, "Detection of image region-duplication with rotation and scaling tolerance," in *ICCCI (1)*, ser. Lecture Notes in Computer Science, J.-S. Pan, S.-M. Chen, and N. T. Nguyen, Eds., vol. 6421. Springer, 2010, pp. 100–108.

[150] H. Huang, W. Guo, and Y. Zhang, "Detection of copy-move forgery in digital images using sift algorithm," in *PACIIA (2)*. IEEE Computer Society, 2008, pp. 272–276.

[151] D. G. Lowe, "Object recognition from local scale-invariant features," in *ICCV*, 1999, pp. 1150–1157.

[152] J. S. Beis and D. G. Lowe, "Shape indexing using approximate nearest-neighbour search in high-dimensional spaces," in *CVPR*, 1997, pp. 1000–1006.

[153] I. Amerini, L. Ballan, R. Caldelli, A. D. Bimbo, and G. Serra, "Geometric tampering estimation by means of a sift-based forensic analysis," in *ICASSP*. IEEE, 2010, pp. 1702–1705.

[154] X. Pan and S. Lyu, "Detecting image region duplication using sift features," in *ICASSP*. IEEE, 2010, pp. 1706–1709.

[155] S.-J. Ryu, M.-J. Lee, and H.-K. Lee, "Detection of copy-rotate-move forgery using zernike moments," in *Information Hiding*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2010, vol. 6387, pp. 51–65.

[156] F. Zernike, "Beugungstheorie des schneideverfahrens und seiner verbesserten form der phasenkontrastmethode," 1934, pp. 689–704, zernike Polynome, orhtogonale Kreisflächen Polynome.

[157] S. Bravo-Solorio and A. K. Nandi, "Automated detection and localisation of duplicated regions affected by reflection, rotation and scaling in image forensics," *Signal Processing*, vol. 91, no. 8, pp. 1759–1770, 2011.

[158] A. E. Dirik and N. D. Memon, "Image tamper detection based on demosaicing artifacts," in *ICIP*. IEEE, 2009, pp. 1497–1500.

[159] Y.-F. Hsu and S.-F. Chang, "Camera response functions for image forensics: An automatic algorithm for splicing detection," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 816–825, 2010. [Online]. Available: http://dx.doi.org/10.1109/TIFS.2010.2077628

[160] W. Luo, Z. Qu, J. Huang, and G. Qiu, "A novel method for detecting cropped and recompressed image block," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 2, april 2007, pp. II–217 –II–220.

[161] T. Bianchi and A. Piva, "Detection of non-aligned double jpeg compression with estimation of primary compression parameters," in *Accepted at ICIP*, 2011.

[162] H. Farid, "Exposing digital forgeries from JPEG ghosts," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 154–160, 2009.

[163] Z. C. Lin, J. F. He, X. Tang, and C. K. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognition*, vol. 42, no. 11, pp. 2492–2501, Nov. 2009. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2009.03.019.

[164] T. Bianchi, A. De Rosa, and A. Piva, "Improved dct coefficient analysis for forgery localization in jpeg images," in *ICASSP*. IEEE, 2011, pp. 2444–2447.

[165] Y.-L. Chen and C.-T. Hsu, "Detecting recompression of jpeg images via periodicity analysis of compression artifacts for tampering detection," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 396–406, 2011.

[166] I. Avcibas, S. Bayram, N. D. Memon, M. Ramkumar, and B. Sankur, "A classifier design for detecting image manipulations," in *ICIP*, 2004, pp. 2645–2648.

[167] W. C. andYun Q. Shi andWei Su, "Image splicing detection using 2-d phase congruency and statistical moments of characteristic function," in *SPIE*, vol. 6505, 2007.

[168] Y. Q. Shi, C. Chen, and W. Chen, "A natural image model approach to splicing detection," in *MM&Sec*, D. Kundur, B. Prabhakaran, J. Dittmann, and J. J. Fridrich, Eds. ACM, 2007, pp. 51–62.

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

[170] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Transactions on Signal Processing*, vol. 53, no. 2-2, pp. 758–767, 2005. [Online]. Available: http://dx.doi.org/10.1109/TSP.2004.839932(410)53

[171] B. Mahdian and S. Saic, "Blind authentication using periodic properties of interpolation," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 529–538, 2008.

[172] N. Dalgaard, C. Mosquera, and F. Pérez-González, "On the role of differentiation for resampling detection," in *ICIP*. IEEE, 2010, pp. 1753–1756.

[173] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *MM&Sec*, A. D. Ker, J. Dittmann, and J. J. Fridrich, Eds. ACM, 2008, pp. 11–20.

[174] M. Kirchner and J. J. Fridrich, "On detection of median filtering in digital images," in *Media Forensics and Security*, ser. SPIE Proceedings, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7541. SPIE, 2010, p. 754110.

[175] M. C. Stamm and K. J. R. Liu, "Blind forensics of contrast enhancement in digital images," in *ICIP*. IEEE, 2008, pp. 3112–3115.

[176] ——, "Forensic estimation and reconstruction of a contrast enhancement mapping," in *ICASSP*. IEEE, 2010, pp. 1698–1701.

[177] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, no. 3, p. 10, 2007.

[178] A. Sarkar, L. Nataraj, and B. S. Manjunath, "Detection of seam carving and localization of seam insertions in digital images," in *11th ACM Workshop on Multimedia and Security*, Sep 2009, pp. 107–116.

[179] C. Fillion and G. Sharma, "Detecting content adaptive scaling of images for forensic applications," in *Media Forensics and Security*, ser. SPIE Proceedings, N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., vol. 7541. SPIE, 2010, p. 75410.

[180] H. Farid, "The lee harvey oswald backyard photos: real or fake?" 2009.

[181] W. Zhang, X. Cao, J. Zhang, J. Zhu, and P. Wang, "Detecting photographic composites using shadows," in *ICME*. IEEE, 2009, pp. 1042–1045.

[182] M. K. Johnson and H. Farid, "Exposing digital forgeries through chromatic aberration," in *MM&Sec*, S. Voloshynovskiy, J. Dittmann, and J. J. Fridrich, Eds. ACM, 2006, pp. 48–55.

[183] I. Yerushalmy and H. Hel-Or, "Digital image forgery detection based on lens and sensor aberration," *International Journal of Computer Vision*, vol. 92, no. 1, pp. 71–91, 2011.

[184] V. Conotter, G. Boato, and H. Farid, "Detecting photo manipulation on signs and billboards," in *ICIP*. IEEE, 2010, pp. 1741–1744.

[185] P. Kakar, N. Sudha, and W. Ser, "Exposing digital image forgeries by detecting discrepancies in motion blur," *IEEE Transactions on Multimedia*, vol. 13, no. 3, pp. 443–452, 2011.

[186] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting duplication," in *MM&Sec*, D. Kundur, B. Prabhakaran, J. Dittmann, and J. J. Fridrich, Eds. ACM, 2007, pp. 35–42.

[187] E. D. Castro and C. Morandi, "Registration of translated and rotated images using finite fourier transforms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 700–703, 1987.

[188] C.-C. Hsu, T.-Y. Hung, C.-W. Lin, and C.-T. Hsu, "Video forgery detection using correlation of noise residue," in *MMSP*. IEEE Signal Processing Society, 2008, pp. 170–174.

[189] W. Wang and H. Farid, "Exposing digital forgeries in interlaced and deinterlaced video," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3-1, pp. 438–449, 2007.

[190] S. Bayram, H. T. Sencar, and N. D. Memon, "Video copy detection based on source device characteristics: a complementary approach to content-based methods," in *Proceedings of the 1st ACM SIGMM International Conference on Multimedia Information Retrieval, MIR 2008, Vancouver, British Columbia, Canada, October 30-31, 2008*, M. S. Lew, A. D. Bimbo, and E. M. Bakker, Eds. ACM, 2008, pp. 435–442. [Online]. Available: http://doi.acm.org/10.1145/1460096.1460167

[191] J. Zhang, Y. Su, and M. Zhang, "Exposing digital video forgery by ghost shadow artifact," in *Proceedings of the First ACM workshop on Multimedia in forensics*, ser. MiFor '09. New York, NY, USA: ACM, 2009, pp. 49–54. [Online]. Available: http://doi.acm.org/10.1145/1631081.1631093

[192] V. Conotter, "Active and passive multimedia forensics," Ph.D. dissertation, University of Trento, Trento, IT, 2011.

[193] R. C. Maher, "Audio forensic examination," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 84–94, Mar. 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4806208

[194] A. J. Cooper, "Detecting butt-spliced edits in forensic digital audio recordings," in *Proceedings of the 39th International AES Conference Audio Forensics: Practices and Challenges*, Hillerød, Denmark, Jun. 2010, pp. 11–21. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=15484

[195] N. Nitanda, M. Haseyama, and H. Kitajima, "Audio-cut detection and audio-segment classification using fuzzy c-means clustering," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 4. Montreal, Quebec, Canada: IEEE, 2004, pp. 325–328. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1326829

[196] M. Kyperountas, C. Kotropoulos, and I. Pitas, "Enhanced eigen-audioframes for audiovisual scene change detection," *IEEE Transactions on Multimedia*, vol. 9, no. 4, pp. 785–797, Jun. 2007. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4202595

[197] R. Yang, Z. Qu, and J. Huang, "Detecting digital audio forgeries by checking frame offsets," in *Proceedings of the 10th ACM Workshop on Multimedia and Security (MM&Sec)*. New York, New York, USA: ACM Press, 2008, pp. 21–26. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1411328.1411334

[198] C. Grigoras, "Digital audio recording analysis: the electric network frequency (enf) criterion," *International Journal of Speech, Language and the Law*, vol. 12, no. 1, pp. 63–76, Jun. 2005. [Online]. Available: http://www.equinoxjournals.com/ojs/index.php/IJSLL/article/view/525

[199] D. Nicolalde and J. Apolinario, "Evaluating digital audio authenticity with spectral distances and enf phase change," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2. Taipei, Taiwan: IEEE, 2009, pp. 1417–1420. [Online]. Available: http://www.computer.org/portal/web/csdl/doi/10.1109/ICASSP.2009.4959859

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

[201] D. Barchiesi and J. Reiss, "Reverse engineering of a mix," *Journal of Audio Engineering Society*, vol. 58, no. 7/8, pp. 563–576, 2010. [Online]. Available: http://goblin.elec.qmul.ac.uk/people/josh/documents/BarchiesiReiss-ReverseEngineeringofaMix.pdf

[202] B. A. Kolasinski, "A framework for automatic mixing using timbral similarity measures and genetic optimization," in *Proceedings of the 124th Convention of the Audio Engineering Society (AES)*, vol. 7496, Amsterdam, The Netherlands, 2008, pp. 1–8. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14626

[203] H. Farid, "Detecting digital forgeries using bispectral analysis," MIT, Cambridge, MA, AI Memo, 1999, [Online]. [Online]. Available: http://hdl.handle.net/1721.1/6678

[204] M. Stein, "Automatic detection of multiple, cascaded audio effects in guitar recordings," in *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx)*, Graz, Austria, 2010, pp. 1–4.

[205] K. Brandenburg, C. Dittmar, M. Gruhne, J. Abeer, H. Lukashevich, P. Dunker, D. Gärtner, W. Kay, S. Nowak, and H. Grossmann, "Music search and recommendation," in *Handbook of Multimedia for Digital Entertainment and Arts*, B. Furht, Ed. Springer, 2009.

[206] G. Peeters and X. Rodet, "Hierarchical gaussian tree with inertia ratio maximization for the classification of large musical instruments databases," in *Proc. of the 6th Int. Conf. on Digital Audio Effects, London*. Citeseer, 2003, pp. 1–6. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.63.1008\&rep=rep1\&type=pdf

[207] A. R. Webb, *Statistical Pattern Recognition*, 2nd ed. Chichester, UK: Wiley, 2002.

[208] *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010, 14-19 March 2010, Sheraton Dallas Hotel, Dallas, Texas, USA*. IEEE, 2010.

[209] *International Symposium on Circuits and Systems (ISCAS 2010), May 30 - June 2, 2010, Paris, France*. IEEE, 2010.

[210] *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2008, March 30 - April 4, 2008, Caesars Palace, Las Vegas, Nevada, USA*. IEEE, 2008.

[211] *International Workshop on Multimedia Signal Processing, MMSP 2008, October 8-10, 2008, Shangri-la Hotel, Cairns, Queensland, Australia*. IEEE Signal Processing Society, 2008.

[212] *Proceedings of the International Conference on Image Processing, ICIP 2010, September 26-29, Hong Kong, China*. IEEE, 2010.

[213] *Proceedings of the International Conference on Image Processing, ICIP 2007, September 16-19, 2007, San Antonio, Texas, USA*. IEEE, 2007.

[214] *Proceedings of the International Conference on Image Processing, ICIP 2009, 7-10 November 2009, Cairo, Egypt*. IEEE, 2009.

[215] *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2009, 19-24 April 2009, Taipei, Taiwan*. IEEE, 2009.

[216] *Proceedings of the International Conference on Image Processing, ICIP 2008, October 12-15, 2008, San Diego, California, USA*. IEEE, 2008.

[217] N. D. Memon, J. Dittmann, A. M. Alattar, and E. J. Delp, Eds., *Media Forensics and Security II, part of the IS&T-SPIE Electronic Imaging Symposium, San Jose, CA, USA, January 18-20, 2010, Proceedings*, ser. SPIE Proceedings, vol. 7541. SPIE, 2010.

[218] H.-J. Kim, Y. Q. Shi, and M. Barni, Eds., *Digital Watermarking - 9th International Workshop, IWDW 2010, Seoul, Korea, October 1-3, 2010, Revised Selected Papers*, ser. Lecture Notes in Computer Science, vol. 6526. Springer, 2011.

[219] S. Voloshynovskiy, J. Dittmann, and J. J. Fridrich, Eds., *Proceedings of the 8th workshop on Multimedia & Security, MM&Sec 2006, Geneva, Switzerland, September 26-27, 2006*. ACM, 2006.

[220] E. J. Delp, J. Dittmann, N. D. Memon, and P. W. Wong, Eds., *Media Forensics and Security I, part of the IS&T-SPIE Electronic Imaging Symposium, San Jose, CA, USA, January 19, 2009, Proceedings*, ser. SPIE Proceedings, vol. 7254. SPIE, 2009.

[221] *Proceedings of the 2007 IEEE International Conference on Multimedia and Expo, ICME 2007, July 2-5, 2007, Beijing, China*. IEEE, 2007.

[222] D. Kundur, B. Prabhakaran, J. Dittmann, and J. J. Fridrich, Eds., *Proceedings of the 9th workshop on Multimedia & Security, MM&Sec 2007, Dallas, Texas, USA, September 20-21, 2007*. ACM, 2007.