Graduate School ETD Form 9
(Revised 12/07)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Nitin Khanna

Entitled Forensic Characterization of Image Capture Devices

For the degree of Doctor of Philosophy

Is approved by the final examining committee:

Edward J. Delp
Chair

George T. Chiu

Mark R. Bell

Jan P. Allebach

To the best of my knowledge and as understood by the student in the *Research Integrity and Copyright Disclaimer (Graduate School Form 20)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): Edward J. Delp

Approved by: V. Balakrishnan 09/18/09
Head of the Graduate Program Date

Graduate School Form 20
(Revised 10/07)

PURDUE UNIVERSITY
GRADUATE SCHOOL

Research Integrity and Copyright Disclaimer

Title of Thesis/Dissertation: Forensic Characterization of Image Capture Devices

For the degree of Doctor of Philosophy

I certify that in the preparation of this thesis, I have observed the provisions of *Purdue University Executive Memorandum No. C-22,* September 6, 1991, *Policy on Integrity in Research.**

Further, I certify that this work is free of plagiarism and all materials appearing in this thesis/dissertation have been properly quoted and attributed.

I certify that all copyrighted material incorporated into this thesis/dissertation is in compliance with the United States' copyright law and that I have received written permission from the copyright owners for my use of their work, which is beyond the scope of the law. I agree to indemnify and save harmless Purdue University from any and all claims that may be asserted or that may arise from any copyright violation.

Nitin Khanna
Signature of Candidate

09/18/09
Date

*Located at http://www.purdue.edu/policies/pages/teach_res_outreach/c_22.html

FORENSIC CHARACTERIZATION OF IMAGE CAPTURE DEVICES

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Nitin Khanna

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

December 2009

Purdue University

West Lafayette, Indiana

This Thesis is dedicated to my Teachers.

Without their training and love, this document would not have been written.

ACKNOWLEDGMENTS

During the past four years at Purdue University, graduate school has been an exciting and challenging experience. Although any amount of writing can not express true gratitude, I would like to take this opportunity to thank those who have provided support, encouragement, and friendship.

I am grateful to my advisor Professor Edward J. Delp for his ongoing encouragement, challenging approach to learning and for his confidence in me. I wish to express my sincere thanks to his broad range of expertise and repeated appeals for me to give up "undergraduate thinking" and "think like a scholar" which has finally come to realization, though the journey has seemed tough at times. He has been instrumental in bringing about many improvements in both the work and the presentation and in my overall attitude. It has been a great honor to be a part of the Video and Image Processing (VIPER) lab. I would like to thank National Science Foundation for supporting this research.

I would like to thank my committee members, Professor Jan P. Allebach, Professor Mark R. Bell and Professor George T.-C. Chiu for their advice, encouragement and insights despite their extremely busy schedules.

I am very fortunate to work with a diverse group of enthusiastic intellectuals, my lab-mates and friends at Purdue University: Dr. Liang Liang, Dr. Limin Liu, Dr. Anthony Frank Martone, Golnaz Abdollahian, Marc Bosch, Ying Chen, Kevin S. Lorenz, Ashok Mariappan, Anand Mariappan, Aravind Mikkilineni, Ka Ki Ng, Oriol Guitart Pla, Francisco Serrano, Deen King-Smith, Satyam Srivastava, Carlos Wang, Fengging (Maggie) Zhu. Special thanks to Aravind Mikkilineni for his constructive suggestions on my research and extraordinary patience in helping me with many computer related issues.

I would like to thank all my friends from Indian Institute of Technology and Purdue University. Their friendship along the journey made my undergraduate and graduate years a very cherishable memory. I am certainly blessed to have so many true friends in the age when even one true friend is hard to find.

I would especially like to thank all my family for their love and support. They have made many sacrifices throughout the years I have been away from home so that I can pursue my academic career. I thank my parents for giving me life and the opportunity to view the world with true perspective.

This material is based upon work supported by the National Science Foundation under Grant No. CNS-0524540. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

Table Page

Table Page

LIST OF FIGURES

Figure Page

ABBREVIATIONS

ADC	Analog-to-Digital Converter
CCD	Charge Coupled Device
CCFL	Cold Cathode Fluorescent Lamp
CFA	Color Filter Array
CIS	Contact Image Sensor
CG	Camera Generated
CMOS	Complementary Metal Oxide Semiconductor
DCT	Discrete Cosine Transform
DPI	Dots Per Inch
DWT	Discrete Wavelet Transform
EM	Expectation Maximization
FMTG	Forensic Monkey Text Generator
FPN	Fixed Pattern Noise
GGD	Generalized Gaussian Distribution
GLDH	Gray-Level Difference Histogram
JPEG	Joint Photographic Experts Group
IQM	Image Quality Measures
K-NN	K-Nearest Neighbor
LDA	Linear Discriminant Analysis
LPA-ICI	Local Polynomial Approximation-Intersection of Confidence Intervals
MSE	Mean Square Error
NCC	Normalized Cross Correlation
OSH	Optimum Separating Hyperplane

PCA	Principle Component Analysis
PMT	Photo-Multiplier Tube
PRNU	Photoresponse Nonuniformity
PRCG	Photo-Realistic Computer Generated
RBF	Radial Basis Function
ROI	region of Interest
RGB	Red-Green-Blue
SVM	Support Vector Machine
SG	Scanner Generated
TIFF	Tagged Image File Format
YMCG	Yellow-Magenta-Cyan-Green

ABSTRACT

Khanna, Nitin. Ph.D., Purdue University, December, 2009. Forensic Characterization of Image Capture Devices. Major Professor: Edward J. Delp.

Forensic characterization of sensors or devices is important in many applications such as establishing the trust and verifying authenticity of data produced by a sensor or device and the sensor or device that created it. Recently there has been a great deal of interest using features intrinsic to a data-generating sensor for the purpose of source identification. Numerous methods have been proposed for various problems related to sensor forensics in general and image forensics in particular. Although a considerable amount of work has been done in forensic identification of digital cameras, more work needs to be done in forensic characterization of scanners, video cameras and other audio devices.

This thesis is aimed at developing tools for forensic characterization of devices or sensors, in particular image capture devices. Statistical feature based classifiers are designed for imaging sensor classification and for source scanner identification for images acquired using flatbed desktop scanners. The methods are based on using imaging sensor pattern noise for scanned photographs and texture features for scanned documents, as device fingerprints. The statistical feature vector based methods provide high accuracies, both for native resolution and lower resolution scanned images. The proposed method perform well with images that have undergone JPEG compression with low quality factors, image sharpening, and contrast stretching. The proposed features are also robust to the scan area used for a particular scan so knowledge of the exact location of scanner's bed used for scanning is not needed.

The sensor noise based source scanner identification scheme is extended for forgery detection in scanned photographs scanned at native resolution of the scanners. This

method can be an effective tool for forgery detection in scanned images if used in co-ordination with other existing methods for forgery detection.

The techniques used for both camera and scanner identification are dependent on having prior knowledge of the class of devices (cameras or scanners) that generated the image. If the image was generated by a digital camera, then the digital camera identification methods must be used. Similarly if the image was generated by a scanner, the scanner identification methods must be used to obtain the best identification results. Use of the sensor pattern noise for classifying digital images based on their originating mechanism, a scanner or a digital camera or the use of computer graphics, is investigated. To achieve this, differences in the characteristics of the sensor noise are used. These differences arise between the two classes due to inherent mechanical differences between their respective sensors and image generation mechanisms. As shown by our results, the proposed scheme does not need the availability of the actual source device for training purposes. Thus, images generated by a completely unknown scanner or digital camera can be classified properly.

1. INTRODUCTION

Advances in digital imaging technologies have led to the development of low-cost and high-resolution digital cameras and scanners, both of which are becoming ubiquitous. Digital images generated by various sources are widely used in a number of applications from medical imaging and law enforcement to banking and daily consumer use [1–5]. The increasing functionality of image editing tools allows even an amateur to easily manipulate images. In some cases a digitally scanned image can meet the threshold definition requirements of a "legal duplicate" if the document can be properly authenticated [6]. There has also been tremendous growth in the areas of computer graphics and computer vision. Growth in these fields combined with the availability of less costly and faster computers have led to the development of software tools which are not only capable of generating photo-realistic images but which can be easily used by a novice. As these technologies advance, it will become easier to create computer generated images which are almost impossible to differentiate from real photographs. These advancements in the area of digital imaging have direct impact on the way our society perceives and uses digital images.

Forensic tools that help establish the origin, authenticity, and chain of custody of digital images are essential to a forensic examiner. These tools can prove to be vital whenever questions of digital image integrity are raised. Some major applications are in, the use of scanned checks [6], use of digital images as evidence in the court [7] or application of child pornography prevention act and its modification [8]. Therefore, a reliable and objective way to examine digital image authenticity is needed. This is different from simply securing the data being sent across the network because we are also authenticating the sensor that is creating the data. One technique that is used to authenticate a device involves embedding information, or a watermark, into the signal generated by the device. This strategy has potential problems in that

the watermark could be attacked, allowing untrusted data to appear authentic. The chances of attack further increase with increase in delay between generation of sensor data and embedding of watermark, which is controlled by the user in most situations.

Digital cameras, scanners, and software tools (such as "3D studio max" and "Maya") are three main sources of digital images. A digital image can originate from a single source or it can be a mosaic made by combining images from more than one source. An image generated by merging a digital photo of a person with a background generated in Photoshop [9] is an example of image belonging to a mixed class: cameras + computers. Similarly, other classes of forged images exist. There are various levels at which the image forensic problems can be addressed. One may want to find the particular device (digital camera or scanner) which generated the image or one might be interested in knowing only the make and model of the device or one may just want to know which source class (camera, scanner or computer generated) the image comes from. In other applications one is interested to know the confidence level with which an image belongs to a claimed source. As summarized in [10, 11], a number of robust methods have been proposed for source camera identification [12–21]. In [22–24], techniques for classification of images based on their sources: scanner, camera and computer generated images, are presented. There have been advances in source scanner identification using sensor noise in the past year. In [25], a direct extension of the sensor noise based source camera identification algorithm [15] was used for source scanner identification. Another approach for scanner model identification using sensor pattern noise is presented in [26]. This method is aimed at classifying images depending upon the scanner model that generated it and not the exact scanner.

The techniques used for both source camera and scanner identification are dependent upon having prior knowledge of the class of device (cameras or scanners). If the image was generated by a digital camera, then the digital camera identification methods must be used. Similarly, if the image was generated by a scanner, the scanner identification methods must be used to obtain the best identification results. Present

day computer generated photo-realistic images are difficult to distinguish from digital camera images if we rely only on human visual system. Hence, in this dissertation, we target two problems. First is that of source scanner identification for scanned images. The second problem is that of ascertaining the class of an image before the source identification appropriate for that class of images is done. The three classes are:

1. Digital Camera Generated (CG) images,
2. Scanner Generated (SG) images, and
3. Photo-Realistic Computer Generated (PRCG) images

We will first define the area of device forensics followed by a description of the image formation systems discussed in this thesis. A brief overview of the state of the art in this area and our approach to the problem is then presented.

1.1 Device Forensics

Device forensics deals with identifying the type, make, model, configuration, and other characteristics of a sensor or device based on observation of the data that the sensor or device produces [27]. The characteristics that uniquely identify the device are known as device signatures.

Given a digital image, the goal of image forensics is to determine the device that created it, whether the image is authentic, or whether the image has been tampered. Determination of regions of manipulations for counterfeit images is also desired. There are various levels at which the image source identification problem can be addressed. One may want to find the particular device (digital camera or scanner) which generated the image or one might be interested in knowing only the make and model of the device.

1.2 Image Generation Systems

There are three primary ways in which digital images can be generated: a digital camera, a scanner and computer graphics tools. In this section, a high level overview of these image generation systems is presented. This is critical in understanding how to distinguish between different image sources.

1.2.1 Digital Camera Architecture

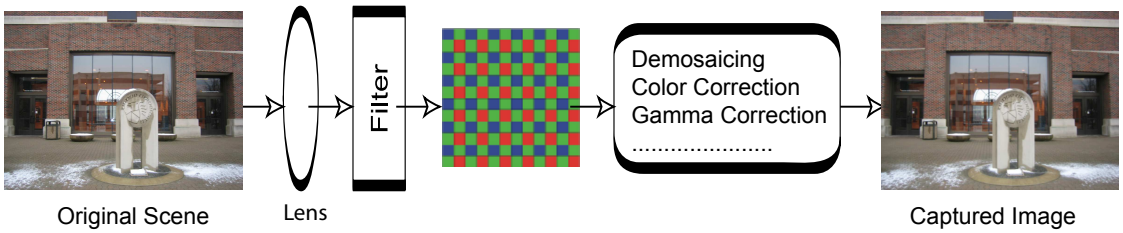


Fig. 1.1. Imaging Pipeline For a Digital Camera.

Basic elements of the digital camera imaging pipeline are shown in Figure 1.1. Even though the exact design details change from manufacturer to manufacturer or model to model, the basic structure of a digital camera pipeline remains the same [28, 29].

First, light from a scene enters the camera through a lens and passes through a set of filters including an anti-aliasing filter. Next the light is "captured" by a sensor. These sensors, typically CCD or CMOS imaging sensors, are color blind in the sense that each pixel captures only intensity information from the light hitting it. To capture color information, the light first passes through a color filter array (CFA) which assigns each pixel on the sensor one of the three (or four) colors. Shown in Figure 1.2 are CFA patterns using RGB and YMCG color spaces, respectively, for a 4×4 block of pixels. The individual color planes are filled in by interpolation using the sampled pixel values. There are a number of different interpolation algorithms which

may be used, depending on the manufacturer. There are also Foveon X3 sensor based cameras (such as the Sigma SD9, Sigma DP1, Polaroid x530) which independently capture all three colors at each pixel location.

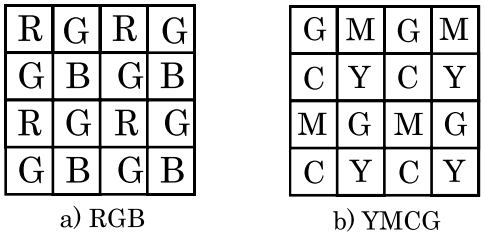


Fig. 1.2. CFA Patterns.

Next, a number of operations are performed by the camera which include white point correction and gamma correction. The image is finally written into the camera memory in a user-specified image format (e.g. RAW, TIFF or JPEG). Although these operations and stages are standard in a digital camera pipeline, the exact processing details in each stage vary from one manufacturer to another, and even between different camera models from the same manufacturer. These variations from one camera to another can be used to determine the camera used to acquire a specific image and the common characteristics of image formation system can be used to differentiate camera generated images from other two classes of images.

1.2.2 Scanner Architecture

The basic architecture of a typical flatbed scanner is shown in Figure 1.3 [30, 31]. A hard copy document is placed face-down on a glass window of the scanner bed and the acquisition process starts. The imaging pipeline for a typical flatbed scanner is shown in Figure 1.4. Using a series of mirrors and lenses, light reflected by the printed patterns are reflected to a photosensitive element that converts it into electrical signals. To complete the process, electrical signals produced by the

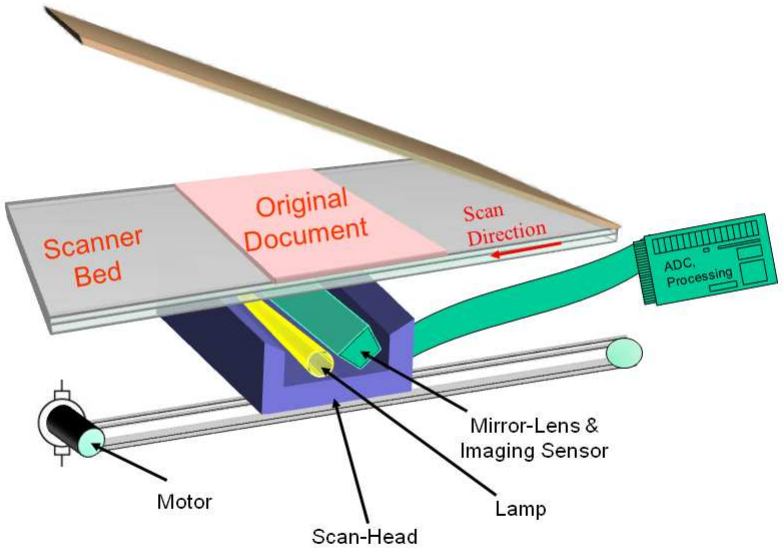


Fig. 1.3. Flatbed Scanner Architecture.

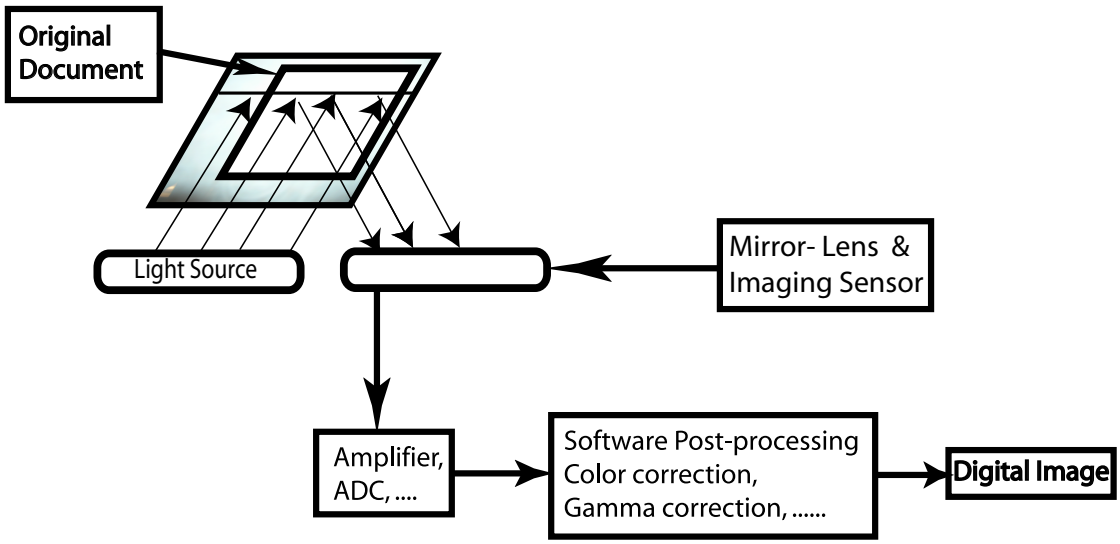


Fig. 1.4. Flatbed Scanner Imaging Pipeline.

sensor are digitized by an analog-to-digital converter (ADC) and are sent to the host computer (Figure 1.5) [1].

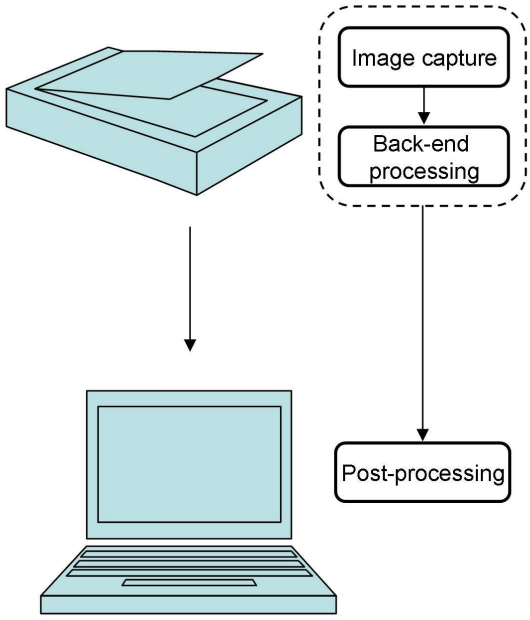


Fig. 1.5. Block Diagram of Operations for a Typical Scanner.

The lamp used to illuminate the document is either a cold cathode fluorescent lamp (CCFL), xenon lamp, or LEDs, while older scanners may use a standard fluorescent lamp. Using a stabilizer belt and a stepper motor, the scan head slowly translates linearly to capture the image. The purpose of the stabilizer bar is to ensure that there is no wobble or deviation in the scan head with respect to the document. Velocity fluctuations in the constant speed portion of the motor's motion may lead to color registration errors in the scanned document [32]. The scan head includes a set of lenses, mirrors, filters, and the imaging sensor. Most desktop scanners use charge-coupled device (CCD) imaging sensors. Other scanners use complementary metal-oxide semiconductor (CMOS) imaging sensors, contact image sensors (CIS), or photomultiplier tube (PMTs) [30, 31].

The native resolution of the scanner is determined by the horizontal and vertical resolution. The number of elements in the linear CCD sensor determines the horizontal optical resolution. The step size of the motor controlling the scan head and the sensor data retrieval time determines the vertical resolution. There are two basic methods for scanning an image at a resolution lower than the hardware resolution of the scanner. One approach is to sub-sample the imaging sensor and read measurements at required pixels only. For example, to produce a 600 DPI scan on a 1200 DPI scanner, the scanner would only sample every other sensor pixel. Another approach involves scanning at the full resolution of the sensor and then down-sampling the results in the scanner's memory. Most good quality scanners adopt the second method since it yields far more accurate results.

1.2.3 Photo-Realistic Computer Generated (PRCG) Images

Realistic image synthesis is important for applications such as simulation, design and advertising [3]. Ferweda et al. [33] define three types of realism for computer graphics: physical realism (the same visual stimulation as the scene), photorealism (same visual response as the scene) and functional realism (same visual information as the scene such as an object's shape and depth). Of the three, photorealistic computer graphics is of special interest to the image forensics community. Photorealism results from various visual effects contained within a 3-D scene, such as those arising from complexity in the scene and object geometry, illumination and the object reflectance of the scene. The two important components of photorealistic graphics synthesis are 1) scene modeling, which includes the modeling of the illumination, object reflectance, and object geometry in a scene; and 2) scene rendering. With realistic scene modeling and correct light-transport simulation, photorealistic computer graphics can be generated.

Scene and Object Modeling

Image-based models are currently used for scene modeling because they can accurately capture the complexity of a real-world scene. Realistic scene illumination can be measured as an environment map using a mirror sphere [34]. This environment map can then be used to model a complex light source in computer graphics rendering. Complex reflectance of real-world surfaces can be modeled by measuring the reflectance from real surface samples. For instance, spatially varying surface reflectance (texture) can be measured from multiple-view photographs [35]. In computer graphics pipeline, the geometry of objects is often represented as a polygonal mesh. This can be obtained via range scanning.

Computer Graphics Rendering

Computer graphics rendering is the process that simulates the light transport between the illumination sources and object surfaces. This light transport may involve multiple bounces of light from one location of the scene to the others that give rise to visual effects such as soft shadows, color bleeding and so on. Current PRCG rendering methods such as ray tracing and radiosity simulate the multiple light bounces between surfaces to produce the global illumination effects of ten features now-a-days in 3D rendering software such as Autodesk 3D Max Studio. Finally, after an image is rendered, it may be processed via a simplified camera model (e.g. only gamma correction) in order to produce a photographic appearance [3].

1.2.4 Sensor Noise

The process of manufacturing imaging sensors introduces various defects which create noise in the pixel values [28, 36]. Sensor noise, which is of interest for use in forensic characterization, can be described in three forms, depending upon its impact on final pixel values and procedures employed to correct it. The first type of noise is

caused by array defects. These include point defects, hot point defects, dead pixels, pixel traps, column defects, and cluster defects. These defects cause pixel values in the image to deviate greatly. For example, dead pixels show up as black in the image and hot point defects show up as very bright pixels in the image, regardless of image content. The second type of noise is pattern noise, which refers to any spatial pattern that does not change significantly from image to image. Pattern noise is caused by dark current and photoresponse nonuniformity (PRNU). Dark currents are stray currents from the sensor substrate into the individual pixels. This varies from pixel to pixel and the variation is known as fixed pattern noise (FPN). FPN is caused by differences in detector size, doping density, and foreign matter trapped during fabrication. PRNU is the variation in pixel responsivity and is present when the device is illuminated. This noise is caused by variations between pixels such as detector size, spectral response, thickness in coatings and other imperfections created during the manufacturing process. The third type of noise is random noise components which vary from frame to frame. This random noise is inevitable and cannot be removed by calibration. However, its statistical characteristics may give some clues about the source imaging device. The first type of noise leads to large deviations in pixel values and is easily corrected in most of the devices available in market. The second type of noise does not lead to large variations in pixel values, and algorithms (such as flat-fielding) used to correct it are difficult to implement in-device. Due to the difficulties in achieving a uniform sensor illumination inside the camera, most consumer cameras do not flat-field their images [2, 15, 17].

The pattern noise can be used for imaging sensor identification. But, it is extremely difficult to obtain the fixed component of the sensor noise by direct methods such as flat fielding. This is due to the fact that in most of the general purpose cameras and scanners, the raw sensor data is unavailable. Also, transforming voltage sensed by the imaging sensor to the output image in JPEG or TIF format requires many complex (non-linear) image processing operations. So, in absence of any direct method, indirect methods of estimating the sensor noise are used.

1.3 Overview of The Dissertation

1.3.1 Contributions

In this dissertation, we studied several new approaches for source scanner identification and image source classification [1,10,11,21–23,37–45]. The main contributions of this dissertation are:

- **Verification of Sensor Noise-based Camera Identification Scheme**: As a first step towards development of new methods for different problems in scanner forensics, we performed extensive experiments for verification of sensor noise based camera forensic method [2, 15, 17]. The results of these independently conducted experiments on a completely different set of cameras, were similar to those reported in earlier papers.

- **Source Scanner Identification from Scanned Images**: We investigated the use of imaging sensor pattern noise for source scanner identification and compared the end-to-end system performance with other existing methods. Our results show that the statistical feature vector based method gives high accuracy for source scanner identification, both for native resolution and lower resolution scanned images. Also, it is possible to discriminate between scanners of the same make and model for images scanned at native scanning resolution. For images scanned at lower non-native resolutions such as 200 DPI, the proposed scheme successfully identifies the scanner make and model, and groups scanners of the same make and model into a single class. The proposed scheme performs well even with images that have undergone JPEG compression with low quality factors, image sharpening, and contrast stretching.

- **Forgery Detection in Scanned Images**: We also extended the use of statistical features of image sensor pattern noise for forgery detection in scanned images and show the efficacy of this method for identifying forgeries in images scanned at native resolution of the scanners. The limitation on minimum size

of forged regions that can be identified with this approach depends upon the size of sliding window. To maintain the statistical significance of the features used for classification, we can not use window sizes below a certain threshold, which was experimentally determined for our datasets. The proposed method identifies the forgeries independent of the image content and fails for the forgeries made by copying and pasting regions within the same image. It can be an effective tool for forgery detection in scanned images, if used in co-ordination with other existing methods for forgery detection.

- **Source Scanner Identification from Scanned Documents**: We proposed methods for source scanner identification for scanned text documents using texture features. The proposed method is robust to JPEG compression and successfully classifies text documents. The proposed features are also robust to the scan area used for a particular scan so we do not need to know which exact location of scanner's bed was used for scanning.

- **Imaging Source Classification**: Use of the sensor pattern noise for classifying digital images based on their originating mechanism: a scanner or a digital camera or a computer graphics algorithm, is investigated. The proposed scheme utilizes statistical properties of the residual noise and the difference in the geometry of the imaging sensors and demonstrates promising results. It does not need the availability of the actual source device for training purposes. Thus, even images generated by a completely unknown scanner or digital camera can be classified properly.

1.3.2 Organization

The primary objective of this dissertation is to develop signal processing tools for image forensics and use them for source scanner identification and image source classification.

Chapter 1 motivates the area of image forensics followed by a brief description of three specific research problems addressed in this dissertation. Each of these is presented in a separate chapter. Each chapter is self-contained with problem statement, methods and results.

Chapter 2 surveys the previous work on image source identification and image source classification. Since our proposed techniques for scanned images are based on sensor noise characterization, therefore we also present detailed experimental results of verification of sensor noise based camera forensic method [2, 15, 17] (Section 2.1.3).

Chapter 3 focuses on source scanner identification for scanned photographs and describes correlation-based and statistical feature-based approaches for this problem. Then extensive experimental results for these two approaches are presented. This is followed by the description of and experimental results for a forgery detection algorithm for scanned images.

Chapter 4 describes the source scanner identification techniques to text documents. We describe the Graylevel Co-Occurrence Matrix (GLCM) features and how the edge color transitions are modeled for text characters.

Chapter 5 describes image source classification and the selection of features relevant for this problem. We provide experimental results on native and non-native resolution scanned images and also analyze the effect of JPEG compression on the classification performance. The main conclusions of this research and future work are finally discussed in Chapter 6.

2. LITERATURE REVIEW

This chapter surveys the existing literature on image source identification and image source classification. Our proposed techniques for scanned images are based on sensor noise characterization, therefore we also present detailed experimental results of verification of state-of-the-art sensor noise-based camera forensic method [2, 15, 17] (Section 2.1.3).

2.1 Image Source Identification

This section presents a brief overview of existing techniques for image source identification. Since there is not much in the literature on source scanner identification, and solutions for source camera identification are closely related to the former, different camera identification techniques are described. These techniques for image source identification can be broadly divided into three sub-categories depending upon the type of features used for the device fingerprint.

2.1.1 Image Features

In [18] and [46], techniques are proposed which use classifiers to determine the source camera using a set of content independent features extracted from the image. The feature vector is constructed from average pixel values, RGB pair correlations, center of mass distributions, RGB pair energy ratios, wavelet based features, and a blind image quality metric. This technique is shown to provide close to 90% classification accuracy across 5 different cameras [18]. Similar results were later reported by Tsai et al. [47] by implementing this scheme on a different set of cameras. Further experiments need to be done to determine whether this method is capable of

distinguishing between similar camera models or between cameras of the exact same model. Also, the large number of images needed to train a classifier for each camera may not always be available. Similar feature based classifiers are applied for source cell phone camera identification in [48, 49].

2.1.2 CFA and Demosaicing Artifacts

Most of the consumer quality digital cameras use a single imaging sensor (either CCD or CMOS) with a color filter array (Figure 1.2) for capturing the image. At each pixel location, the sensor captures information for only one of the colors. To obtain the full color image, the other two (or three in case of YMCG color filters) colors have to be estimated by interpolation or demosaicing techniques. This interpolation introduces correlations between the samples of a color image. The non-interpolated samples are unlikely to be correlated in the same way as the interpolated samples. There are a number of interpolation algorithms which may be used. The interpolation artifacts produced are dependent on the interpolation technique used. Suitable features can be designed to capture these differences in interpolation artifacts. Hence, the features based on detection of demosaicing artifacts can be used for image source identification. One common difficulty faced by these methods is that many of the interpolation techniques used are non-linear and image content dependent. After the CFA interpolation, non-linear (such as gamma correction) and lossy (such as JPEG compression) operations are performed to produce the final image and in general one does not have access to the raw images.

In [20,50], a method is proposed based on the observation that both the size of the interpolation kernel and the demosaicing algorithm vary from camera to camera. The source camera of a digital image is identified based on the estimation of the color interpolation parameters used by the camera. This method is limited to images that are not highly compressed since the compression artifacts suppress and remove the spatial correlation between the pixels created by the CFA interpolation [20,50]. Furthermore,

the interpolation operation is highly non-linear, making it strongly dependent on the nature of the scene. These methods are fine-tuned to prevent visual artifacts such as over-smoothed edges or poor color transitions in busy parts of the image. In smooth regions of the image these algorithms exhibit a more linear characteristic. Therefore, smooth and nonsmooth regions of images are treated separately [20]. Since no a priori information is assumed on the size of interpolation kernel, probability maps are obtained for varying sizes of kernels. When viewed in the frequency domain, these probability maps show peaks at various frequencies with varying magnitudes indicating the structure of correlation between the spatial samples. The classifier relies on two sets of features: the set of weighting coefficients used for interpolation, and the peak locations and magnitudes in the frequency spectrum. A Support Vector Machine (SVM) classifier is used to test the effectiveness of the proposed features.

A similar technique, presented in [19], assumes a linear model for the periodic correlations introduced by CFA interpolation. The assumption is that each interpolated pixel is correlated to a weighted sum of pixels in a small neighborhood centered about itself. While perhaps overly simplistic when compared to the highly nonlinear nature of most CFA interpolation algorithms, this simple model is both easy to parameterize and can reasonably approximate the CFA interpolation algorithms. Note that most CFA algorithms estimate a missing color sample from neighboring samples in all three color channels. For simplicity, however this technique ignores these inter channel correlations and treats each color channel independently. In practice, neither the specific form of the correlations (that is, the parameters of the linear model) nor which samples are correlated to their neighbors are known. To estimate these both simultaneously, the expectation maximization (EM) algorithm is used [51].

2.1.3 Sensor Based Characterization

Other methods for digital camera identification are based on characterizing the imaging sensor used in the device. In [12], it is shown that defective pixels can be used

for reliable camera identification even from lossy compressed images. This type of noise, generated by hot or dead pixels, is typically more prevalent in cheap cameras. The noise can be visualized by averaging multiple images from the same camera. These errors can remain visible after the image is compressed. Many cameras post-process the captured image to remove these types of noise, so this technique cannot always be used.

Fridrich et al. did pioneering work in developing source camera identification techniques using the imaging sensor's pattern noise [2, 13–17]. The identification is based on pixel nonuniformity noise which is a unique stochastic characteristic of both charge coupled device (CCD) and complementary metal oxide semiconductor (CMOS) imaging sensors. Reliable identification is possible even from images that are resampled and JPEG compressed. The pattern noise is caused by several factors such as pixel non-uniformity, dust specks on the optics, optical interference, and dark current [28, 36]. The high frequency part of the pattern noise is estimated by subtracting a denoised version of the image from the original. This is performed using a wavelet-based denoising filter [52]. A camera's reference pattern is determined by averaging the noise patterns from multiple images obtained from the camera. The reference pattern serves as an intrinsic signature of the camera. To identify the source camera, the noise pattern from an image is correlated with known reference patterns from a set of cameras, and the camera corresponding to the reference pattern giving maximum correlation is chosen to be the source camera. In [16, 17], an improved method for source camera identification based on joint estimation and detection of the camera photo-response non-uniformity (PRNU) in images is presented. This scheme is extended in [53] for detection of forgery in digital camera images. Some assumptions made in this technique are open for questioning. The wavelet denoising filter [52], for example, assumes that the image in the wavelet domain is a non-stationary Gaussian process and that the pattern noise is a stationary Gaussian process. Since these assumptions are satisfied only approximately, the pattern noise extracted using the denoising filter is not Gaussian. Another problem is that the filter is applied to the

image on slightly overlapping blocks and it pads image borders with zeros. This leads to a small residual dependence between all extracted noise patterns. Furthermore, reference patterns from different cameras are often slightly correlated due to the use of similar or even the same image processing methods.

There have been advances in source scanner identification using sensor noise in the past year. In [25], a direct extension of the sensor noise based source camera identification technique [15] was used for source scanner identification. Experiments were performed on five scanners of three different models and four digital cameras. Images were scanned at the native scanner resolution (1200 DPI) as well as at a lower non-native resolution (200 DPI) and stored as uncompressed (TIFF) color images. The reference patterns were generated by averaging noise patterns from 100 training images. All the experiments performed on scanned images have shown lower classification accuracy compared to similar experiments for source camera identification. It has been shown that using a 1-dimensional reference pattern gives better classification accuracy on images scanned at non-native resolution while the 2-dimensional reference pattern gives better results on images scanned at native resolution of the scanners. This is due to the predominance of local disturbances such as dust specs and scratches on the glass plate in the 2-dimensional reference patterns of the scanners which are suppressed in the 1-dimensional reference pattern through averaging over multiple scan lines. Further experiments are needed to determine the robustness of this scheme when such local disturbances are present for two reasons. First, the dust specs or other temporary disturbances on the glass plate are easily changed due to cleaning and other factors. Second, the presence of other permanent disturbances such as scratches on the glass plate will vary depending upon which portion of the scanner bed is used for scanning the image. Further experiments show that one possible reason for the observed decline in performance is post-processing operations such as better denoising techniques including flat-fielding and heavy down-sampling [25].

Another approach for scanner model identification using sensor pattern noise described in [26] uses three sets of features extracted from each scanned image. This

method is aimed at classifying images depending upon the scanner model that generated it and not the exact scanner. Experiments were performed on 26 images scanned at 150 DPI from seven different scanners. Training on 13 images and testing on 13 images gives a 90% average classification accuracy, and the leave-one-out scenario gives a 96% average classification accuracy. Since the dimensionality of the feature vectors used with the SVM classifier is 25, further testing on a larger database need to be performed to obtain more conclusive results. The performance of this scheme has to be further tested on images obtained from multiple scanners of the same model. Furthermore, it has been shown that the classification scheme using statistical features of the sensor noise performs much better than those using high-order wavelet statistics or image quality metrics, which give an average classification accuracy of 77% and 68% respectively. Again, further experiments need to be performed on a larger image database to test the effectiveness of image quality metrics based and high-order wavelet statistics based schemes, which have 45-dimensional and 216-dimensional feature vectors respectively.

The next three sections describe our study for verification of sensor noise based source camera identification [13]. This study on an independent dataset and implementation was done as a first step towards designing sensor noise based approaches for source scanner identification.

Correlation based approaches

Figures 2.1 and 2.2 show the training and testing protocols used in [13] for source camera identification using sensor pattern noise. As in [15] a wavelet based denoising filter [52] is used for denoising the image. This denoising filter needs standard deviation of the noise as an input parameter, which is chosen to be 5. A camera's reference pattern is determined by averaging the noise patterns from multiple images captured by the camera. This reference pattern serves as an intrinsic signature of the camera (Figure 2.1). To identify the source camera, the noise pattern from an image

is correlated with known reference patterns from a set of cameras (Figure 2.2). The camera corresponding to the reference pattern with highest correlation is chosen to be the source camera [15].

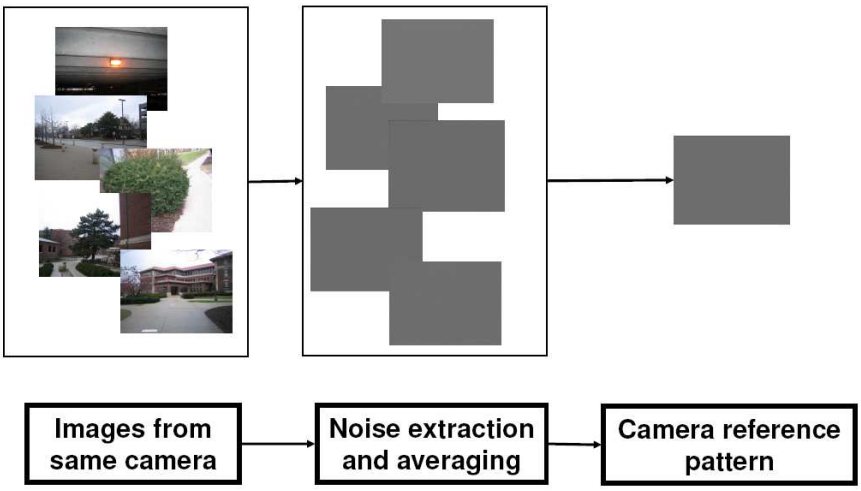


Fig. 2.1. Classifier Training for Correlation-Based Approach.

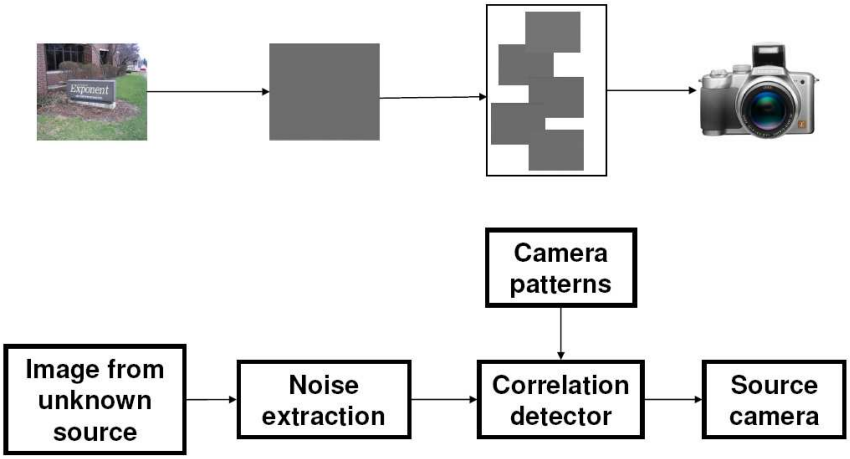


Fig. 2.2. Source Camera Identification Using a Correlation-Based Detection Scheme.

Let I^k denote the k^{th} input image of size $M \times N$ pixels (M rows and N columns). Let I^k_{noise} be the noise corresponding to the original input image I^k and let $I^k_{denoised}$ be the output of the denoising filter. Then as in [15],

$$I^k_{noise} = I^k - I^k_{denoised} \tag{2.1}$$

Let K be the number of images used to obtain the reference pattern of a particular digital camera. Then the 2-dimensional array reference pattern is obtained as

$$\widetilde{I}^{array}_{noise}(i,j) = \frac{1}{K}\sum_{k=1}^{K} I^k_{noise}(i,j); \quad 1 \leq i \leq M \ , 1 \leq j \leq N \tag{2.2}$$

Correlation is used as a measure of the similarity between the camera reference patterns and the noise pattern of a given image [15]. Correlation between two vectors $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^N$ is defined as

$$\mathbf{C}(X,Y) = \frac{(X - \bar{X}) \cdot (Y - \bar{Y})}{||X - \bar{X}|| \, . \, ||Y - \bar{Y}||} \tag{2.3}$$

This correlation is used for source camera identification from an unknown image. The camera corresponding to the reference pattern giving highest correlation is decided as the source camera. An experimental threshold can also be determined, then camera corresponding to the reference pattern giving correlation value higher then the threshold will be decided as the source camera.

Experimental Results

Table 2.1 lists the digital still cameras used in our experiments. Each of these cameras are used to capture images at various resolutions and image quality settings, with all other settings left to default, such as auto focus, red eye correction and white balance. Images taken by these cameras have similar as well as dissimilar contents. Figure 2.3 shows a sample of the images used in this study.

Table 2.1
Camera Set Used for Evaluation of Method for Camera Identification

	Device	Sensor Size (inch)	Sensor Resolution	Maximum Picture Size	Image Format
$c1$	Canon PowerShot SD200-1	1/2.5	3.2 MP	2048 x 1536	JPEG
$c2$	Canon PowerShot SD200-2	1/2.5	3.2 MP	2048 x 1536	JPEG
$c3$	Nikon Coolpix 7600	1/1.8	7.1 MP	3072 x 2304	JPEG
$c4$	Panasonic DMC-FZ20	1/2.5	5 MP	2560 x 1920	JPEG/TIFF
$c5$	Nikon Coolpix 4100	1/2.5	4 MP	2288 x 1712	JPEG
$c6$	Nokia 6630(3G smartphone)			1280 x 960	JPEG
$c7$	Olympus E-10	2/3	4 MP	2240 x 1680	JPEG/TIFF
$c8$	Olympus D-360L			1280 x 960	JPEG/TIFF
$c9$	Panasonic Lumix DMC-FZ4-1	1/2.5	4 MP	2304 x 1728	JPEG/TIFF
$c10$	Panasonic Lumix DMC-FZ4-2	1/2.5	4 MP	2304 x 1728	JPEG/TIFF

Fig. 2.3. Sample Images Used in Our Study.

Reference Camera Pattern Generation

Reference Camera patterns are obtained by averaging the noise extracted from multiple images from the same camera 3.2. To achieve this it is not necessary to have that camera in our possession as only the training images are needed and no

internal design parameters need to be accessed. To determine the optimal number of training images needed to generate the camera reference pattern, 20 randomly chosen images are used as test images and the average correlations (ρ_{avg}) between the camera reference pattern generated from N_p training images and these testing images are plotted in Figure 2.4.

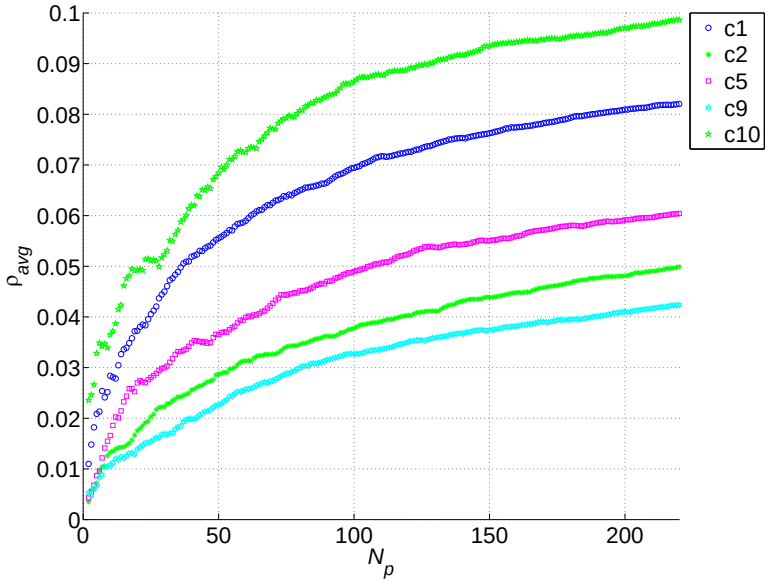


Fig. 2.4. Average correlation ρ_{avg} as a Function of the Number of Images N_p Used for Estimating the Reference Pattern.

As the correlation detector is highly sensitive to geometrical transformations such as rotation and given an unknown image one does not know in which way the user held the camera, we need to incorporate these causes of desynchronization before obtaining the correlation. After estimating the noise, it is rotated both +/- 90 degrees and then higher of the two correlations is used.

Image Identification from Unprocessed Images

In these experiments for source camera identification using images of unknown origin, camera reference patterns are estimated using 200 randomly chosen training images. Figures 2.5, 2.6, 2.7, 2.8 and 2.9 show the correlations for various images from a camera with the reference patterns from all other cameras. Eleven reference patterns corresponding to nine different source cameras are used. For camera c_3, two reference patterns are used, one obtained from images captured at resolution 3072×2304 and second one obtained from images captured at resolution 2048×1536. This is to examine the effect of sizes of the reference patterns on source camera identification. In estimating the correlation between noise patterns of different size, the larger of the two is always cropped from the top left corner to match the size of the smaller one. In Section 2.1.3, experiments are done by resizing the image patterns to match the size of the reference patterns. The source camera is chosen based on the reference pattern with the highest correlation value. In all cases the classification accuracy is greater than 98%. The first 200 images correspond to those used for estimation of the reference pattern and rest are used for testing. It is to be noted that even though the correlation between noises from test images and the correct reference pattern is comparatively less than correlation between noises from images used for estimating the reference pattern and the correct reference pattern, the correlation with the correct reference pattern is much higher than with the incorrect reference patterns. The correlation with correct reference pattern is much lower for the images of night sky or those obtained by closing the lid of the camera lens. This observation is consistent with all the cameras.

Effect of JPEG Compression on Image Identification

In this set of experiments, effect of JPEG compression on source camera identification is analyzed. Since the noise extracted using the wavelet based denoising filter corresponds to the high spatial frequencies, the correlation between image noise and

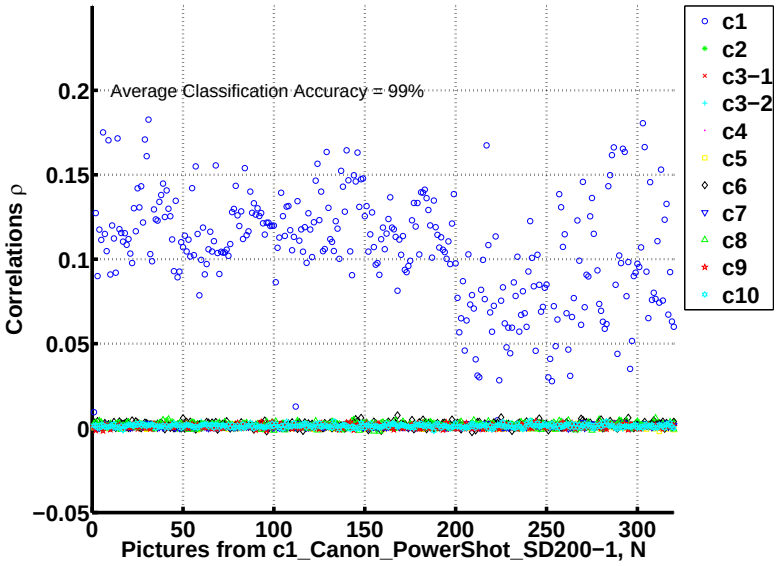


Fig. 2.5. Correlation of Noise from c1 with 11 Reference Patterns.

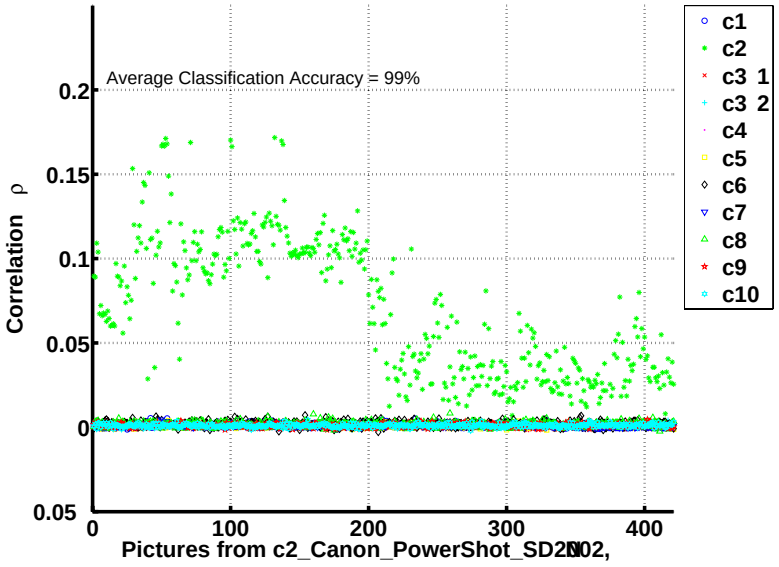


Fig. 2.6. Correlation of Noise from c2 with 11 Reference Patterns.

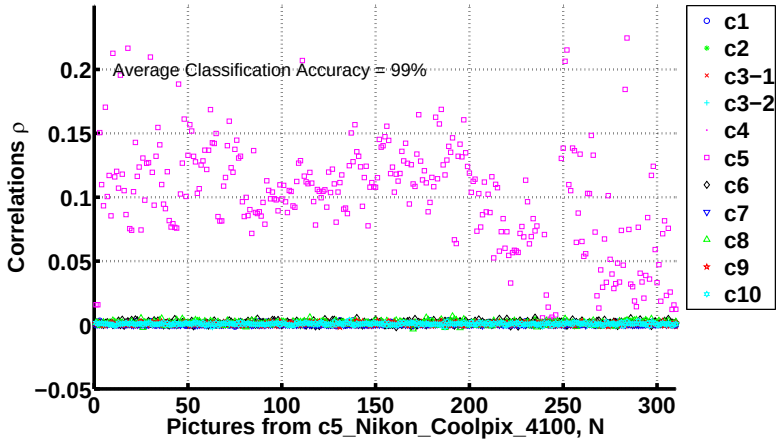


Fig. 2.7. Correlation of Noise from c5 with 11 Reference Patterns.

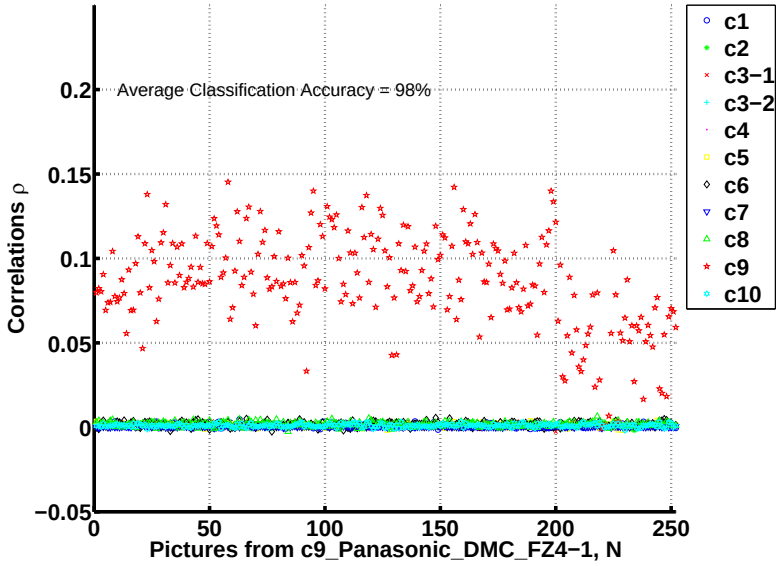


Fig. 2.8. Correlation of Noise from c9 with 11 Reference Patterns.

the reference patterns is expected to decrease. The experiments on different cameras show that this is indeed true. At the same time, correlation with the wrong reference patterns also decreases and accurate source camera identification is still possible.

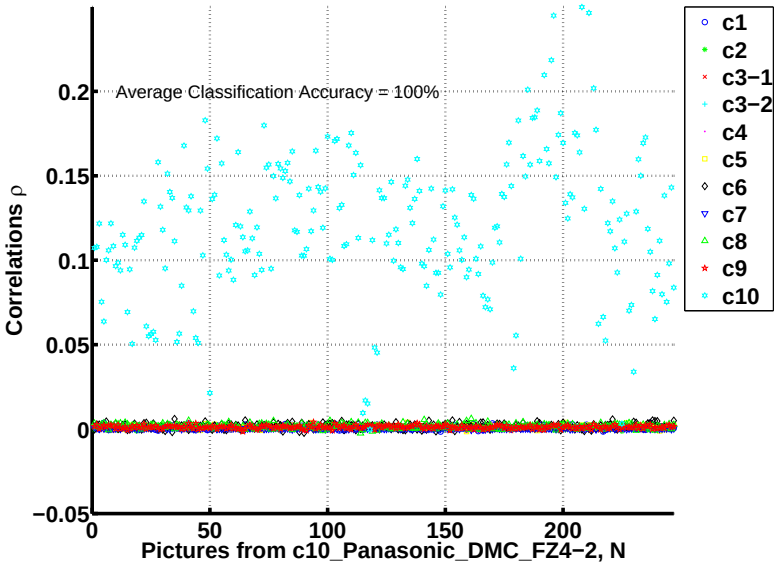


Fig. 2.9. Correlation of Noise from c10 with 11 Reference Patterns.

Figures 2.10, 2.11, 2.12 and 2.13 show the variation in mean and variance of correlation between test images for different JPEG quality factors and the reference patterns from correct and incorrect cameras.

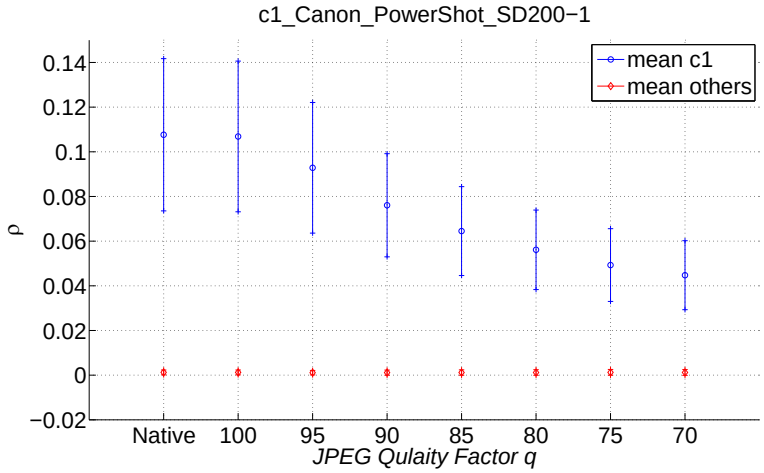


Fig. 2.10. Mean and Standard Deviation of ρ as a Function of the JPEG Quality Factor.

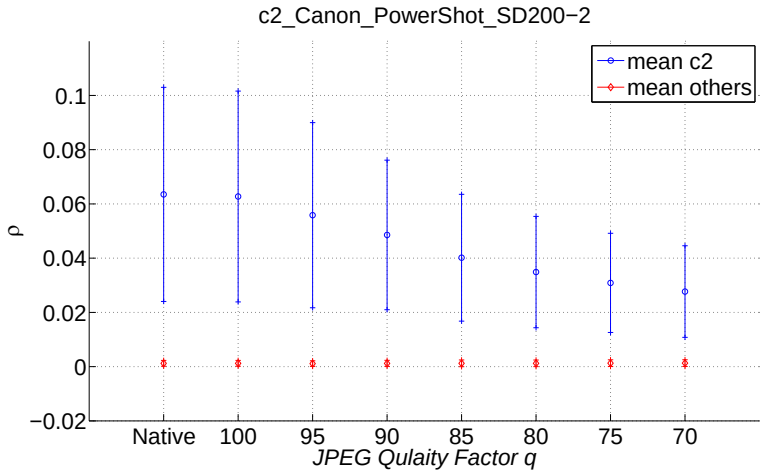


Fig. 2.11. Mean and Standard Deviation of ρ as a Function of the JPEG Quality Factor.

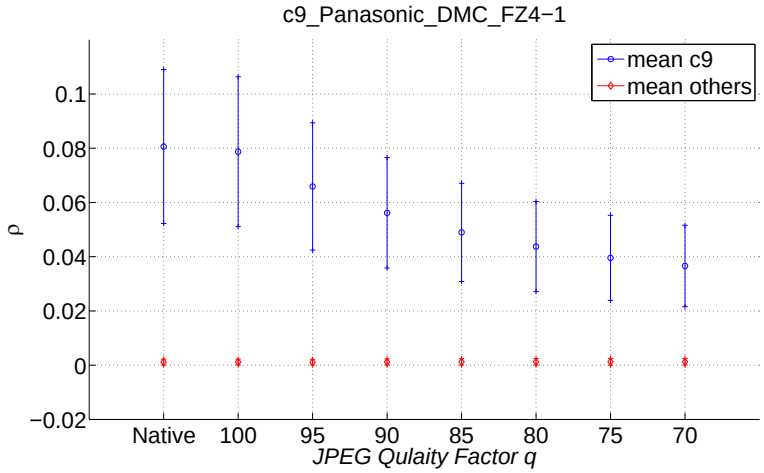


Fig. 2.12. Mean and Standard Deviation of ρ as a Function of the JPEG Quality Factor.

Effect of Resampling On Image Identification

This section investigates the possibility of identifying images obtained at a lower resolution than the maximal resolution. Three hundred fifty images were captured

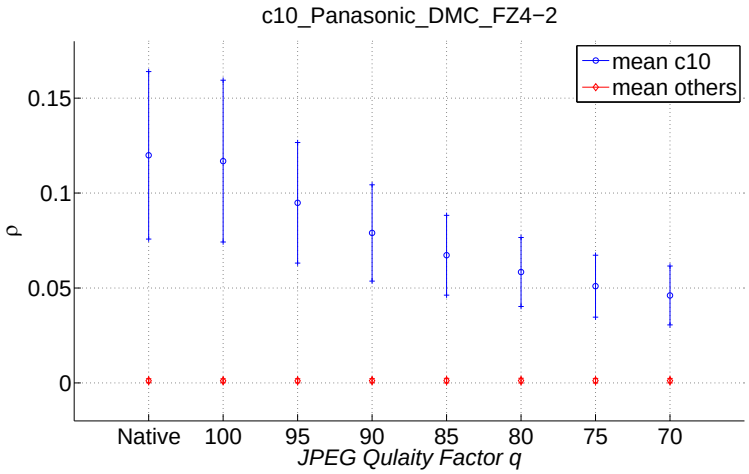


Fig. 2.13. Mean and Standard Deviation of ρ as a Function of the JPEG Quality Factor.

using camera $c1\_Canon\_SD200-1$ at a resolution 1024×768. Assuming that these images have been captured at lower resolution or rescaled in computer but has not been cropped. For source camera identification, the camera patterns estimated earlier from images at maximum resolution of the camera has been used. For the cases when the size of the noise pattern of an unknown image does not match with the size of a reference camera pattern, corresponding image has been resampled using "bicubic" interpolation and then noise extracted from this resampled image is correlated with the known camera reference patterns to decide about the source camera.

As the Figure 2.14 shows, source camera identification is possible even from images captured at a resolution smaller than the maximum allowed by a camera. In Figure 2.14, the reference pattern $c1-1$ is estimated by averaging the first 200 images captured at resolution 1024×768, while the reference pattern $c1-2$ is estimated by averaging 200 images captured at resolution 2048×1536. Since most of the digital cameras do not use simple resampling methods such as "bicubic" interpolation to obtain a lower resolution image, the correlation of noise extracted from a 1024×768 size image is expected to be higher with the reference pattern $c1-1$ (especially for first 200 images as they are used to obtain the reference pattern) than with reference

pattern $c1-2$. This is indeed true in Figure 2.14. The correlation with the reference pattern $c1-2$ is consistently higher than correlation with any other reference pattern and thus classification accuracy is 100%, though with a smaller tolerance and thus less reliable.

Experiments to see the effect of simultaneous application of JPEG compression and resampling show a similar decline in the correlation, and maintain 100% classification accuracy.

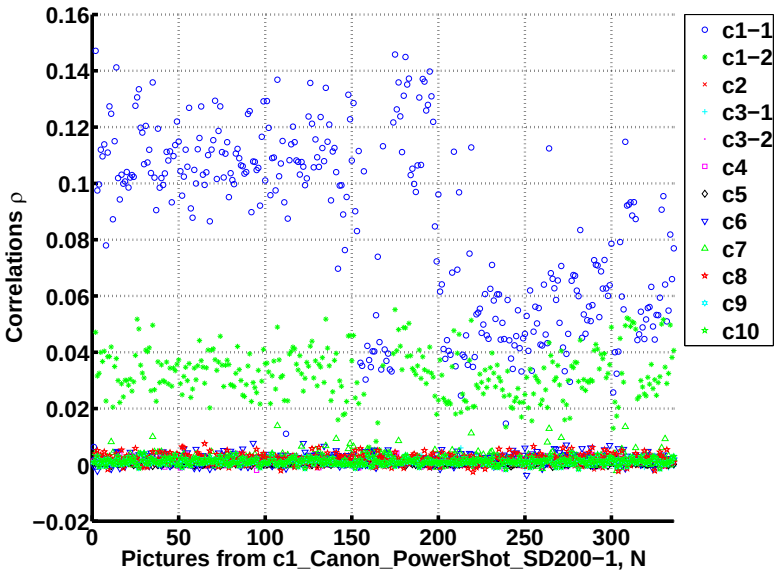


Fig. 2.14. Identification of Low Resolution (1024×768) c1_Canon_SD200-1 Images.

Effect of Malicious Processing

The issue of "preventing" source camera identification by removing the pattern noise from an image is addressed in this section. In the experiments performed here, noise extracted from the denoised image is correlated with the reference patterns obtained from initial training images (as used in the earlier sections). Figures 2.15, 2.16 and 2.17 show the correlations for various denoised images from a camera with the

reference patterns from all other cameras. Comparing with Figures 2.6, 2.8 and 2.9, the correlations for images undergone malicious processing of removing the noise is less than the correlations with non-processed images. Even then the classification accuracy remains greater than 98%.

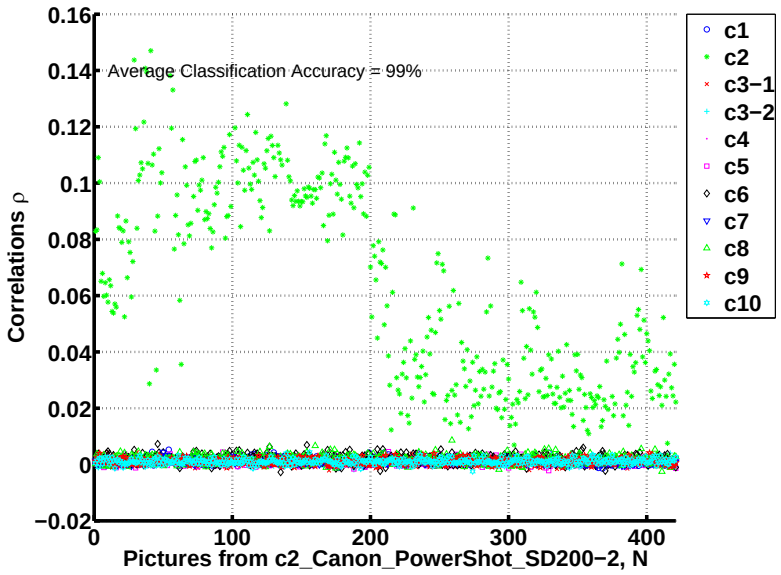


Fig. 2.15. Correlation of Denoised c2_Canon_SD200-2 Images with Reference Patterns from all the Cameras.

2.2 Image Source Classification

The techniques for source identification uniquely assign an image of unknown origin to it's originating device and need training images from that particular device to make appropriate classifier. In some situations, one may not have access to the training images from a particular device and one may just be interested to know the class of image generating system. This section presents a brief overview of existing techniques for image source classification. The aim of these techniques is to classify an image of unknown source as: digital camera generated, scanner generated or photo-

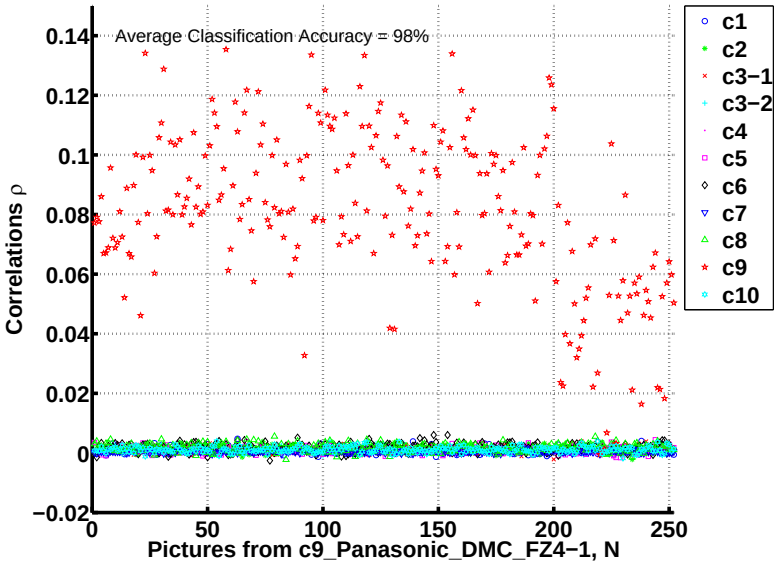


Fig. 2.16. Correlation of Denoised c9_Panasonic_DMC-FZ4-1 Images with Reference Patterns from all the Cameras.

realistic computer generated. These techniques can be broadly divided into three sub-categories depending upon the type of features used as the class fingerprint.

2.2.1 Image Features

Even though many computer generated images are very similar to real images that a human observer will fail to differentiate between the two [54], there are still subtle differences between their statistical properties such as color distribution and wavelet coefficients. These differences can be exploited to extract features which will differentiate real images from PRCG. One common limitation expected with these methods is their inability to identify scanned images since most of the features are image content dependent and not generative process dependent.

In [55] a statistical model based on first and higher-order wavelet statistics is used to differentiate between photo realistic and real images. The features used by

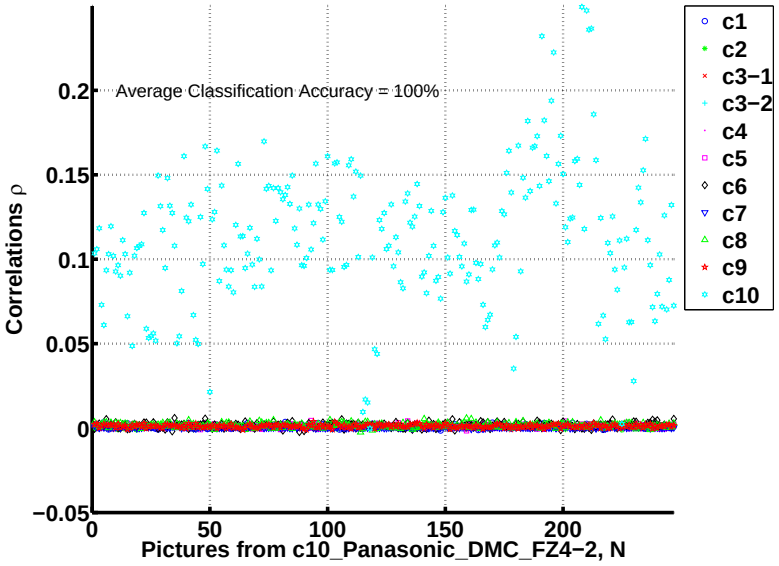


Fig. 2.17. Correlation of Denoised c10_Panasonic_DMC-FZ4-2 Images with Reference Patterns from all the Cameras.

the proposed method are based on the observation that wavelet subband coefficients for natural images typically follow a distribution which is modeled by a generalized Laplacian. Corresponding coefficients of PRCG images are not expected to have similar distribution. Instead of fitting the generalized Laplacian, the first four order statistics are used for statistical modeling. The statistical model consists of two sets of statistics. The first set consists of the first four order statistics (mean, variance, skewness, and kurtosis) of the subband coefficient histograms at each orientation, scale, and color channel (12 values per scale, per color channel). The second set of statistical features is based on the errors in inter-subband linear predictors of coefficient magnitudes at each orientation, scale, and color channel. First four order statistics of each of these error distributions is also taken (12 values per scale, per color channel). Hence, in total for a multi-scale decomposition with scales $i = 1, ..., n$, the total number of coefficient statistics is $36(n-1)$ ($12(n-1)$ per color channel), and the total number of error statistics is also $36(n-1)$, yielding a grand total of $72(n-1)$

statistics. In [55], n is chosen to be 4 and thus for each image a 216-dimensional feature vector is obtained. A non-linear SVM trained on 32,000 photographic images and 4800 PRCG images, and tested on 8000 photographic images and 1200 PRCG images, correctly classifies 66.8% of the photographic images with a 1.2% false negative rate.

Physical differences in the generation of photographic and PRCG images can be modeled using a geometry based approach [56]. A geometry-based image description by means of the fractal geometry is used at the finest scale and differential geometry at the intermediate scale. Further, local patches of the image intensity function are sampled to form a patch distribution. This method extracts a 192-dimensional feature vector from analysis on local patch statistics, local fractal dimension, surface gradient, quadratic geometry and Beltrami flow. The proposed features successfully work on recaptured images as well by incorporating sample recaptured images in the training phase. A SVM classifier built for these features gives an average accuracy of 83.5% as compared to an accuracy of 80.3% for wavelet features on same dataset [56].

The observation that wavelet coefficients of real images and PRCG follow different models is utilized in [57] to discriminate between CG and PRCG images. It is observed that while wavelet coefficients of real images (CG) are modeled by generalized Gaussian distribution (GGD), those of a not-so photo realistic image (image with few objects and lacking visual artifacts and noise generally present in photographic images) are modeled by sum of a Dirac delta function and a GGD. Corresponding coefficients of photorealistic-yet noisy image are modeled by a Cauchy distribution. Similar analysis is earlier used for differentiating real images from steganographic images generated by "adding" information bearing noise to the original images. A three-level wavelet decomposition is performed and the first diagonal band is also decomposed into four parts. Thus, each color image has a total of 48 subbands. Three features are extracted from each of these 48 subbands, resulting in a 144-dimensional feature vector for each of the color image. Feature vector estimation is done by first taking DFT of the normalized histograms of each of the subbands and than filtering these DFTs with two high-pass and one band-pass filter. These three features lie in

different ranges for photographic and PRCG images, for example, the feature obtained from applying band-pass filter have much smaller value for photographic images than for PRCG images and so on. Experiments conducted using a Fisher linear discriminant trained on approximately 2000 images from each of the two classes and tested on different 2000 images from each class, shows the efficacy of the proposed scheme. Images are compressed at JPEG quality 80 or higher. These features not only have slightly better performance, they also take almost half the time than 216-dimensional features used in [55] and almost $1/30^{th}$ of time than physics motivated features used in [56].

Another classifier based approach for detecting differences between CG and PRCG images using visual features derived from color, edge and saturation and texture features extracted with Gabor filters is proposed in [58]. Four visual features used are: the number of unique colors, spatial variation of color, pixel saturation and intensity edges. The use of the number of unique colors is based on the observation that computer generated images tend to have fewer unique colors than real images. Even though present image generation tools on computers provide a large color pallet, as does a real pallet, still computer generated images generally have lesser number of colors intrinsic to the mechanism of image generation. For example, generally an edge line in a natural image is not of exactly the same color while a line generated using a computer graphics tool maintains the exact same color along the complete line unless we do something to change it subsequently. The richness of the color pallet of a image is measured as a ratio of total number of unique RGB triplets in the image to the total number of pixels. The effect of noise can be reduced by counting only those pixels which appear in more than a threshold number of pixels. Spatial variation of color in PRCG images is expected to be less than that in the real image and so it is used as another feature. Pixel saturation is also used as a feature based on the observation that mean and variation of pixel saturation of PRCG are more than those of real images [58]. The number of saturated and unsaturated pixels is obtained by counting the highest bin and the lowest bin in the saturation

histogram. Since the real images generally have more intensity edges than do PRCG, the ratio of number of pure intensity edges to the total number of edge pixels is used as another feature to differentiate real images from PRCG. To represent the homogeneous texture features, mean and standard deviation of the magnitude of the transformed coefficients obtained by applying multi-level Gabor filters are used. The performance analysis is done for three different classifiers, non-linear SVM, weighted k-nearest neighbors and fuzzy k-nearest neighbors on a dataset of around thousand images of each of the two classes; CG and PRCG (with around 200 images for testing and rest of them used for training). Accuracy of 99% for CG and 91.5% for PRCG images is reported for Gabor filter based texture features. Other visual features also show some abilities to perform differentiation [58].

Another method based on statistical features of wavelet coefficients is proposed in [59]. This method uses HSV Color model and statistical moments of characteristic function of the image and wavelet subbands for identifying computer graphics. Although this feature extraction process have some similarities with that in [57]. Two major differences are, 1) the extraction of features in HSV color space instead of RGB color space and 2) using similar features from prediction error image as well. The first three moments of the characteristic function of the histograms of wavelet coefficients (12 subbands obtained from three level decomposition) and the original image, give a 39-dimensional feature per color channel. Similar 39-dimensional features per color channel are also extracted from the prediction error image. Hence for every color image, a 234-dimensional feature vector is obtained. Using a non-linear SVM with the proposed 234-dimensional features extracted from HSV color space, an accuracy of 82.1% is reported on Columbia Image Dataset. This is slightly better than the 80.8% accuracy obtained by [55] on the same database. To obtain further improvements in performance, this method is extended in [60] to use genetic algorithm for selecting the optimal set of features. By using a genetic algorithm, a reduced 100-dimensional feature set is found which performs slightly better than the original 234-dimensional

features. Fractal geometry can also be used for discriminating between PRCG and CG images [61].

2.2.2 CFA and Demosaicing Artifacts

Most consumer digital cameras use a single imaging sensor (either CCD or CMOS) with a color filter array (Figure 1.2) for capturing the image while most of the flatbed scanners either use three different imaging sensors or three different light sources in conjunction with a single imaging sensor. Thus, while for scanned images, at each pixel, each color channel is independently captured, for digital camera images interpolation or demosaicing technique are used to obtain the full color image. This interpolation introduces correlations between the samples of a color image. The non-interpolated samples are unlikely to be correlated in the same way as the interpolated samples. Although, there are a number of different interpolation methods, suitable features can be designed to capture the common artifacts produced by all of these interpolation techniques. PRCG images are not expected to have these demosaicing artifacts. Hence, the features based on detection of demosaicing artifacts can be used for image source classification.

In [62], traces of demosaicing and chromatic aberration are used to differentiate CG from PRCG. Demosaicing features work well for high quality images while chromatic aberration features work well for wide range of compression qualities. The first set of features is based on the detection of the existence of color filter interpolation in the images. Existence of a Bayer pattern is based on measuring the mean squared error (MSE) between the image and re-interpolated versions with different types of CFAs. After taking the minimum over the group of possible CFA patterns, the PRCG images are expected to have significantly larger mean squared error as compared to corresponding error for CG images. For the measurement of mean squared error, an image is divided into $D \times D$ blocks and only the non-smooth blocks (those having standard deviation larger than a certain threshold) are used. Based on the mean

squared error, the CFA pattern giving minimum error for different blocks are selected. The pattern number which yields the second minimum MSE is also used for feature extraction. In the presence of CFA interpolation, these values will not be uniformly distributed. Thus, a measure of uniformity of these two pattern numbers over all non-smooth sub-blocks of image, are the first two CFA features. Next two features are derived from the error metric averaged over all the blocks. The feature set corresponding to chromatic aberration is aimed at detecting misalignment between color channels. This misalignment occurs due to chromatic aberration, variations in reflective index of the optical glass formulation used for manufacturing lenses. The mutual information between color channels is used as a measure of the misalignment between the color channels. When this mutual information is obtained for different values of shift vectors, then the mutual information will attain its maximum for that shift vector which aligns the color channels. Assuming that there is no misalignment between the color channels of PRCG images, the mutual information will be maximized for no shift and will reduce suddenly. For real images, the mutual information will be close to constant for a range of shifts. Hence, the variance of mutual information in a range of shifts is used as the chromatic aberration feature. This feature will have a comparatively high value for PRCG images than for real images. Using a SVM classifier, with 900 images from each of the two classes used for training and another 900 images used for testing, the CFA feature gives an accuracy of 98.1% on high quality images (JPEG quality 95 and 90) as compared to an accuracy of 89.33% achieved by chromatic aberration feature and 99.6% achieved by wavelet features under similar setting. The combination of Bayer features and wavelet features has an accuracy of 99.9%. While the chromatic aberration feature does not give a very high accuracy as compared to other features, it's performance is consistent over a wide range of compressed images. For the combined set compressed at JPEG quality 50 to 100, chromatic aberration feature has an accuracy of 90%.

2.2.3 Sensor Based Characterization

Imaging sensor pattern noise is used as fingerprints of both classes of image capture devices, digital cameras [2, 15, 17] and flatbed scanners [26, 37, 43]. These techniques for source camera or source scanner identification utilize the observation that different image capture devices (cameras or scanners) have unique fingerprints of pattern noise. Although these fingerprints vary from device to device, they may have common statistical properties as the underlying device model, imaging sensor technologies and post-processing operations remain same. Similarly even though a large number of computer graphics tools exists for creating PRCG images, there are similarities among these generative algorithms. Therefore, the residuals of PRCG images share common statistical structures which are different from the statistical structures present in the pattern noises from digital camera and scanner generated images. Hence, this class of methods are based on searching some features of pattern noise which remain same for images from a class of devices and vary amidst different source classes.

The method proposed in [63] is aimed at differentiating digital camera images from computer generated images. Due to the differences in the image generation processes, the residuals obtained from digital camera images exhibit some common characteristics which is lacking in other types of images. The estimation of pattern noise is done in the same way as in [15]. Three reference patterns are estimated from 300 training images of different classes, images from multiple cameras, images created using "Maya" and images created using "3D studio max". Correlation between the reference patterns and residual noise from an unknown image is used for deciding the class of the image. Although there are some differences in reference patterns for the two classes, this method does not give high accuracies.

In [24], statistical properties of noise residuals are jointly used with estimated color interpolation coefficients and corresponding errors to differentiate between images produced by cameras, cell phone cameras, scanners and computer graphics. For extracting the color interpolation based features, assuming the use of a specific color

filter array, the image pixels are divided into three types of regions; region with high vertical gradient, region with high horizontal gradient and smooth region. Linear interpolation coefficients are estimated for each of these three regions. These steps are repeated for different CFA patterns and features are extracted from the CFA pattern giving lowest error. Residual noise features are obtained from image denoising, wavelet analysis and neighborhood prediction. On an image dataset of 100 images from each of the four classes, an average accuracy of 94% is obtained using the leave-one-out method [24].

3. SOURCE SCANNER IDENTIFICATION FROM SCANNED IMAGES

In this chapter we present methods for authenticating images that have been captured by flatbed desktop scanners, using sensor pattern noise. We extend the correlation-based approach used for authenticating digital cameras [15] by using a reference pattern that is one-dimensional instead of two-dimensional. To improve classification accuracy, we incorporate special features of the scanning system, such as the use of one-dimensional image sensor and resulting complexities in using direct extension of digital camera forensic methods. This is done by using a set of statistical features of the sensor noise as scanner signature. The proposed technique uses a SVM classifier to classify the images based upon statistical features obtained from the sensor pattern noise and results in significantly higher accuracy in comparison to correlation-based approaches. Since the sensor pattern noise is estimated using a simple averaging method, further improvements in results may be obtained by using the improved method for sensor noise estimation presented in [16, 17].

In our initial experiments, the proposed set of statistical features are extracted from the pattern noise estimated using a single denoising filter [37]. This scheme gave high classification accuracy for images scanned at native resolution of the scanner but did not work well for heavily down-sampled and post-processed images. Therefore, we extended this scheme to use a denoising filterbank with four denoising filters [43]. This extended scheme works very well for heavily down-sampled and post-processed images also. Our proposed statistical features differ from features used in the sensor noise based scheme of [26], in utilizing special characteristics of the scanner system such as the use of a one dimensional sensor for image capture. Extensive experimentation on a large set of scanners and many different scanning scenarios show the effectiveness of

our proposed scheme. Experiments on images that have undergone post-processing operations such as sharpening and contrast-stretching show that the chosen statistical features survive these operations and allow source scanner identification even after these post-processing operations.

3.1 Correlation Based Approaches

First the high frequency part of the noise is estimated by subtracting a denoised version of an image from the original image [15]. The denoising filter is based on an anisotropic local polynomial estimator [64].

After estimating the noise, the scanner's reference pattern is determined by averaging the noise patterns from multiple scanned images. This reference pattern serves as a signature of the scanner (Figure 3.1). To identify the source scanner of a given image, its estimated noise pattern is correlated with known reference patterns from a set of scanners (Figure 3.2). The scanner corresponding to the reference pattern with the highest correlation is chosen to be the source scanner.

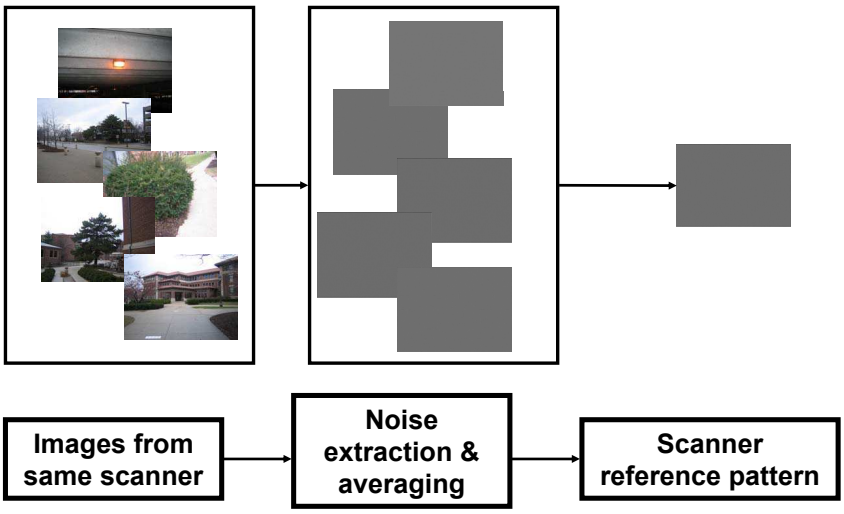


Fig. 3.1. Source Scanner Identification: Classifier Training for Correlation Based Approach.

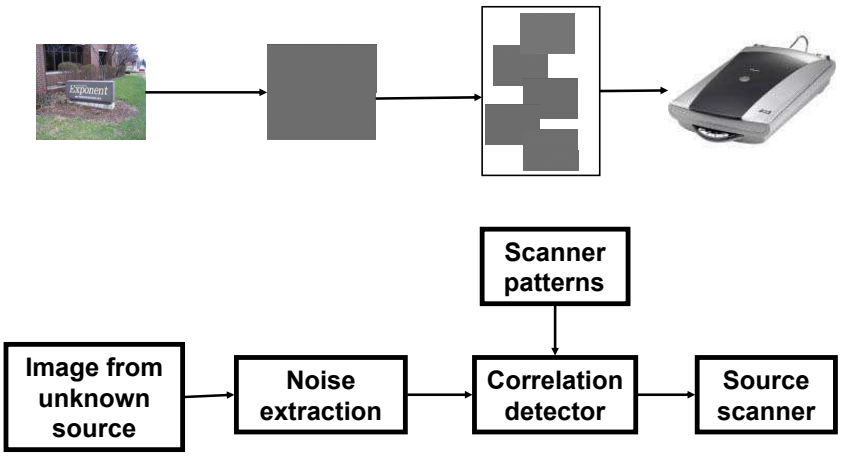


Fig. 3.2. Source Scanner Identification: Classifier Testing for Correlation Based Approach.

In contrast to digital cameras, flatbed scanners use a linear one-dimensional sensor array. Using a one-dimensional version of the two-dimensional array reference pattern, as described in [15], is more appropriate in this case. The linear sensor noise pattern is obtained by averaging all the rows of the noise estimated from an image. The linear sensor reference pattern for a particular scanner is obtained by taking the average of linear sensor noise patterns from multiple images scanned by the same scanner (Figure 3.3). This linear row reference pattern serves as an intrinsic signature of the scanner. To identify the source scanner of an image, its linear noise pattern is correlated with known reference patterns from a set of scanners. The scanner corresponding to the reference pattern with highest correlation is chosen to be the source scanner.

Let I^k denote the k^{th} input image of size $M \times N$ pixels (M rows and N columns). Let I^k_{noise} be the noise corresponding to the original input image I^k and let $I^k_{denoised}$ be the result of applying a denoising filter on I. Then, as in [15],

$$I^k_{noise} = I^k - I^k_{denoised} \tag{3.1}$$

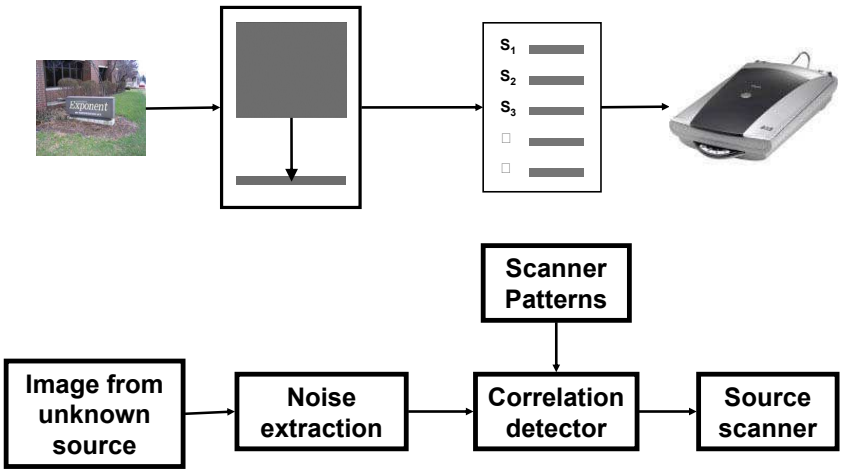


Fig. 3.3. Source Scanner Identification: Correlation Based Detector Using 1-D Row Reference Pattern.

Let K be the number of images used to obtain the reference pattern of a particular scanner. Then the two-dimensional array reference pattern is obtained as

$$\widetilde{I}^{array}_{noise}(i,j) = \frac{1}{K}\sum_{k=1}^{K} I^{k}_{noise}(i,j); \quad 1 \le i \le M \ , 1 \le j \le N \tag{3.2}$$

The linear row reference pattern is obtained as

$$\widetilde{I}^{linear}_{noise}(1,j) = \frac{1}{M}\sum_{i=1}^{M} \widetilde{I}^{array}_{noise}(i,j); \quad 1 \le j \le N \tag{3.3}$$

As explained above, correlation is used as a measure of the similarity between the scanner reference patterns and the noise pattern of a given image [15]. Correlation between two vectors $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^N$ is defined as

$$\mathbf{C}(X,Y) = \frac{(X-\bar{X})\cdot(Y-\bar{Y})}{||X-\bar{X}||||Y-\bar{Y}||} \tag{3.4}$$

This correlation is used to classify scanners. The scanner corresponding to the reference pattern giving highest correlation is chosen as the source scanner. An experimental threshold can also be determined, in which case the scanner corresponding to the reference pattern giving a correlation value higher than the threshold is chosen as the source scanner.

3.2 Statistical Features Based Approach

One of the main differences between the image capturing processes for digital cameras and flatbed scanners is in the usage of sensor elements when capturing an image. Digital cameras use the entire sensor to capture an image, whereas scanners use only a portion of the sensor array determined by location of image on the scanner bed. To correctly estimate scanner reference patterns for correlation-based source scanner identification, all scanned images used for training and testing must be scanned at the exact same location on the scanner bed. Failure to do so will result in comparing noise values from two different sensor locations. This is referred as desynchronization problem faced by correlation detectors. This desynchronization problems comes both in the estimation and the detection of reference patterns. However, this requirement is not typically met in real world scanning scenarios. Hence, the simple approach of correlation detection as used in [15] may not work for flatbed scanners. This is demonstrated in [37] where small images scanned from random locations on the scanner bed are used to estimate the scanner reference pattern.

One way to solve the desynchronization problem for the correlation-based technique is to estimate the scanner reference pattern for the entire scan area. This can be accomplished by using large images, or multiple smaller images tiled across the scanner bed. Detection of the reference pattern in any given image can then be performed using normalized cross correlation (NCC) [65]. The highest value of NCC among all known scanners determines the source scanner as well as the scanning location on the scanner bed.

Implementation of this NCC technique requires storage of large reference patterns, as well as long data-acquisition and computation times. The reference pattern for a flatbed scanner with a native resolution 1200 DPI will be approximately 10800×14400 pixels (500 MBytes) in size. Practical constraints on storage, computation, and data acquisition time motivate the search for alternative techniques for source scanner identification which can make use of smaller training images. Furthermore,

estimation of reference patterns for the complete scanner bed requires possession of all the training devices. A method capable of using a limited number of smaller training images from the same scanner would be ideal. The following sections describe a statistical feature-based technique using support vector machine (SVM) classification which is shown to overcome the stated problems with correlation-based techniques.

Selection of relevant features from the sensor noise is the key to accurate and robust source scanner identification. The features selected should satisfy the following requirements:

- **Independent of image content**
- **Characteristic of the scanner**: Features should capture the characteristics of given scanner, and preferably should differentiate among different scanners of the same make and model
- **Independent of scan area**: Features should be able to characterize the source scanner even if images are placed at different positions on the scanner's glass plate

Scan area independence of proposed features is justified because of following reason. An image scanned twice from different non-overlapping locations on the same scanner will contain different PRNU because the PRNU originates from variations in manufacturing process. The proposed scheme uses sensor noise-based scanner fingerprints. The fixed component of sensor noise is caused by PRNU as well as noise-like characteristics left after post processing steps which include a number of non-linear operations on the values read by sensor array. Thus, the statistical properties of the fixed component of noise are expected to remain same irrespective of the image placement on scanner bed. This is the reason behind using the statistical features of the fixed component of sensor noise for source scanner identification. Our experimental results suggest that this is true.

3.2.1 Statistical Feature Extraction

The scanner scans an image by translating a linear sensor array along the length of its scanning bed. Each row of the resulting digital image is generated by the same set of sensor pixels. Thus, for scanned images the average of all the rows of sensor noise will give an estimate of the fixed "row-pattern". Averaging will reduce the random component and at the same time enhance the fixed component of the noise. In addition to the statistical features along the row direction, features are extracted along the column direction also. This is done in order to compare the two statistics. Let I denote the input image of size $M \times N$ pixels (M rows and N columns) and I_{noise} be the noise corresponding to the image. Let $I_{denoised}$ be the result of applying a denoising filter on I. Then, as in [15],

$$I_{noise} = I - I_{denoised} \tag{3.5}$$

The procedure to extract features from a single color channel is described below. The same procedure is applied to all the three channels separately to get a complete feature vector.

Let $\widetilde{I}^r_{noise}$ and $\widetilde{I}^c_{noise}$ denote the average of all the rows and columns of the noise (I_{noise}) respectively (Equations 3.6 and 3.7).

$$\widetilde{I}^r_{noise}(1,j) = \frac{1}{M}\sum_{i=1}^{M} I_{noise}(i,j);\ \ 1 \le j \le N \tag{3.6}$$

$$\widetilde{I}^c_{noise}(i,1) = \frac{1}{N}\sum_{j=1}^{N} I_{noise}(i,j);\ \ 1 \le i \le M \tag{3.7}$$

Let $\rho_{row}(i)$ denote the correlation value between the average of all the rows ($\widetilde{I}^r_{noise}$) and i^{th} row of the noise (I_{noise}) (Equation 3.8). Similarly, $\rho_{col}(j)$ denotes the value of correlation between the average of all the columns ($\widetilde{I}^c_{noise}$) and j^{th} column of the noise (I_{noise}) (Equation 3.9).

$$\rho_{row}(i) = \mathbf{C}(\widetilde{I}^r_{noise}, I_{noise}(i,.)) \tag{3.8}$$

$$\rho_{col}(j) = \mathbf{C}(\widetilde{I}^{c}_{noise}, I_{noise}(., j)) \tag{3.9}$$

ρ_{row} is expected to have larger values than ρ_{col} since there is a periodicity between rows of the fixed component of the sensor noise of a scanned image (Section 1.2). The statistical properties of ρ_{row}, ρ_{col}, $\widetilde{I}^{r}_{noise}$ and $\widetilde{I}^{c}_{noise}$ capture the essential properties of an image which are useful for discriminating between different scanners. As an example, for a low-quality scanner having large amount of random noise, such as that due to fluctuations in lighting conditions, values of ρ_{row} will be comparatively small and close to the values of ρ_{col}. On the other hand, a high-quality scanner is not expected to have large amount of random noise and thus the values of ρ_{row} is usually much larger than the values of ρ_{col}. Furthermore, for a low-quality scanner, $\widetilde{I}^{r}_{noise}$ and $\widetilde{I}^{c}_{noise}$ will have much higher energy than corresponding values for a high-quality scanner. The mean, standard deviation, skewness and kurtosis of ρ_{row} and ρ_{col} are the first eight features extracted from each color channel of the input image. The standard deviation, skewness and kurtosis of $\widetilde{I}^{r}_{noise}$ and $\widetilde{I}^{c}_{noise}$ correspond to features 9 through 14. The last feature for every channel is given by Equation 3.10 which is a representative of the relative difference in periodicity between the row and column directions of sensor noise. Since we expect ρ_{row} to be large for high-quality scanners and small for low-quality scanners, f_{15} will have a high positive value for all. A few exceptions are the very low-quality scanners and images which have undergone post-processing operation such as very heavy down-sampling or JPEG compression which have a large impact on the sensor noise.

$$f_{15} = \left(1 - \frac{\frac{1}{N}\sum_{j=1}^{N} \rho col(j)}{\frac{1}{M}\sum_{i=1}^{M} \rho_{row}(i)}\right) \times 100 \tag{3.10}$$

By extracting these 15 features from each of the three color channels, a 45 dimensional feature vector is obtained for each scanned image. To capture the three color channels, some scanners use three different linear sensors while others use a single imaging sensor in coordination with a tri-color light source. To capture this difference among scanners of different make and models, six additional features are used.

These features are obtained by taking mutual correlations of $\widetilde{I}^{r}_{noise}$ from different color channels (same for $\widetilde{I}^{c}_{noise}$). Hence, in total each scanned image has a 51 dimensional feature vector associated with it.

In our previous work [37] on source scanner identification from images scanned at native scanner resolution, a recently developed anisotropic local polynomial estimator for image restoration based on directional multiscale optimizations [64] was used for denoising. In this study, a denoising filter bank comprising of four different denoising algorithms: LPA-ICI (Local polynomial approximation - intersection of confidence intervals) denoising scheme [64], median filtering (size 3×3) and Wiener adaptive image denoising for neighborhood sizes 3×3 and 5×5, is used. Using a set of denoising algorithms helps to better capture different types of sensor noise [26]. These denoising algorithms are chosen based on the performance of the complete filter bank in scanner identification. Initial experiments on different linear filtering algorithms such as those using an averaging filter and a Gaussian filter demonstrated that the linear filtering algorithms are not as effective in scanner identification as those used in the proposed scheme. Each denoising algorithm is independently applied to each color band in an image. The features extracted from individual blocks of the filter bank are concatenated to create the final feature vector for each scanned image. Hence, each scanned image has a 204 dimensional feature vector associated with it.

To reduce the dimensionality of the feature vectors, linear discriminant analysis (LDA) [66] is used and a ten dimensional feature vector is obtained for each image. Each component of the ten dimensional feature vector is then a linear combination of the original 204 features. Finally a Support Vector Machine (Appendix A) classifier is used to classify these ten dimensional feature vectors.

3.3 Experimental Results - Correlation Based Methods

Table 3.1 lists the scanners used in our experiments. Experiments are performed on images scanned at the native resolution of the scanners as well as on images scanned

at a lower non-native resolution, such as 200 DPI. Images are generally scanned at a lower resolution to meet constraints on storage space, scanning time, and transmission bandwidth. This adds further complexity to the task of source scanner identification since images scanned at a lower resolution go through heavy down sampling which changes the sensor noise characteristics (Section 1.2).

Table 3.1
Scanner Set Used for Evaluation of Method for Scanner Identification from Scanned Images.

	Make/Model	Sensor	Native Resolution (DPI)
S_1	Epson Perfection 4490 Photo	CCD	4800
S_2	HP ScanJet 6300c-1	CCD	1200
S_3	HP ScanJet 6300c-2	CCD	1200
S_4	HP ScanJet 8250	CCD	4800
S_5	Mustek 1200 III EP	CCD	1200
S_6	Visioneer OneTouch 7300	CIS	1200
S_7	Canon LiDE 25	CIS	1200
S_8	Canon LiDE 70	CIS	1200
S_9	OpticSlim 2420	CIS	1200
S_{10}	Visioneer OneTouch 7100	CCD	1200
S_{11}	Mustek ScanExpress A3	CCD	600

3.3.1 2-D Reference Pattern

To evaluate the effectiveness of the sensor noise based source camera identification technique for source scanner identification, experiments are performed for images scanned at the native resolution of the scanners. These experiments use smaller sub-images of size 1024×768 pixels for taking source scanner identification decisions.

This is done primarily for two reasons: 1) To deal with the large sizes (of the order of 10800×14400 pixels) of images scanned at native resolution and 2) To take into account the lack of information about exact location on the scanner-bed which is used for scanning a particular image. The images scanned at native resolution are sliced into blocks of size 1024×768 pixels. This block size is arbitrarily chosen to provide for statistical significance of features used for classification, reasonable processing time and memory usage. In this experiment, approximately 300 sub-images from each of the four scanners (S_1, S_2, S_3, S_4) are used. One hundred randomly chosen sub-images (from each scanner) are used to estimate the two dimensional array reference patterns. Testing is performed using the remaining sub-images. The anisotropic local polynomial estimator based denoising method (LPA-ICI) [64] is used to estimate the noise in the images and the source scanner is determined using correlation between the estimated 2-D noise and the known reference patterns.

Tables 3.2 and 3.3 show the confusion matrices for classification between pairs of scanners. The $(i, j)^{th}$ entry of the confusion matrix denotes the percentage of sub-images which belong to the i^{th} scanner but are classified as coming from the j^{th} scanner. Using the two dimensional array reference pattern gives an average classification accuracy of 72% and 84.5%, for the scanner pairs (S_1 , S_2) and (S_2 , S_4) respectively.

Table 3.2
Confusion Matrices for Correlation Using 2D Reference Pattern (pair-wise performance, S_1 vs. S_2).

		Predicted	
		S_1	S_2
Actual	S_1	66.8	33.2
	S_2	22.5	77.5

Table 3.3
Confusion Matrices for Correlation Using 2D Reference Pattern (pairwise performance, S_2 vs. S_4).

		Predicted	
		S_2	S_4
Actual	S_2	69.4	30.6
	S_4	0.4	99.6

3.3.2 1-D Reference Pattern

The same images used in Section 3.3.1 are used in this experiment. One hundred randomly chosen sub-images from each scanner are used to estimate the one dimensional row reference patterns. The source class in this case is determined through correlation of the 1-D noise and reference patterns. Tables 3.4 and 3.5 show the confusion matrix for classification between pairs of scanners. Using the one dimensional row reference pattern gives an average classification accuracy of 71% and 92.5%, for the scanner pairs (S_1 , S_2) and (S_2 , S_4) respectively. Other pairs have similar accuracies.

Table 3.4
Confusion Matrices for Correlation Using 1D Reference Pattern (pairwise performance, S_1 vs. S_2).

		Predicted	
		S_1	S_2
Actual	S_1	63.7	36.3
	S_2	21.6	78.4

Tables 3.6 and 3.7 show the confusion matrices for source scanner identification among three scanners by using the two dimensional array reference patterns and one dimensional row reference patterns respectively. For classification among these three

Table 3.5

Confusion Matrices for Correlation Using 1D Reference Pattern (pairwise performance, S_2 vs. S_4).

		Predicted	
		S_2	S_4
Actual	S_2	85.1	14.9
	S_4	0.0	100.0

scanners, using the array reference pattern gives an average classification accuracy of 74% while using the row reference pattern gives an average classification acuracy of 77.6%.

Table 3.6

Confusion Matrix for Correlation Using 2D Reference Pattern (over three scanners).

		Predicted		
		S_1	S_2	S_4
Actual	S_1	63.3	20.8	15.8
	S_2	11.7	59.5	28.8
	S_4	0.4	0.0	99.6

As discussed in Section 3.1, the results presented in this section imply that the row reference pattern provides better results for source scanner identification than the two dimensional array reference pattern. But both of them fall short of achieving our objective of reliable scanner identification.

Table 3.7

Confusion Matrix for Correlation Using 1D Reference Pattern (over three scanners).

		Predicted		
		S_1	S_2	S_4
Actual	S_1	63.3	23.6	13.1
	S_2	10.8	69.4	19.8
	S_4	0.0	0.0	100.0

3.4 Experimental Results - Statistical Features Based Method

The experimental procedure for source scanner identification using statistical features of the sensor noise is shown in Figure 3.4. The `LIBSVM` package [67, 68] is used in this study. Before using the SVM classifier, the features are scaled to the range $[-1, 1]$. The mapping is decided by the values of the features in the training set and the same mapping is applied to the features in the testing set. A radial basis function (RBF) is chosen as the kernel function and a grid search is performed to select the best parameters for the kernel. To generate the final confusion matrices, SVM training and testing steps are repeated multiple times using a random selection of images for the training and testing sets.

3.4.1 Scan Area Independence

Out of the eleven scanners, seven scanners that are representative of the complete set, $S_1, S_2, S_3, S_4, S_6, S_7, S_9$ four CCD and three CIS, with two of the exact same make and model, are used in experiments performed at native resolution. Approximately 40 images are scanned from each of these seven scanners at their respective native scanning resolutions. The scanned images are then sliced into blocks of size 1024×768

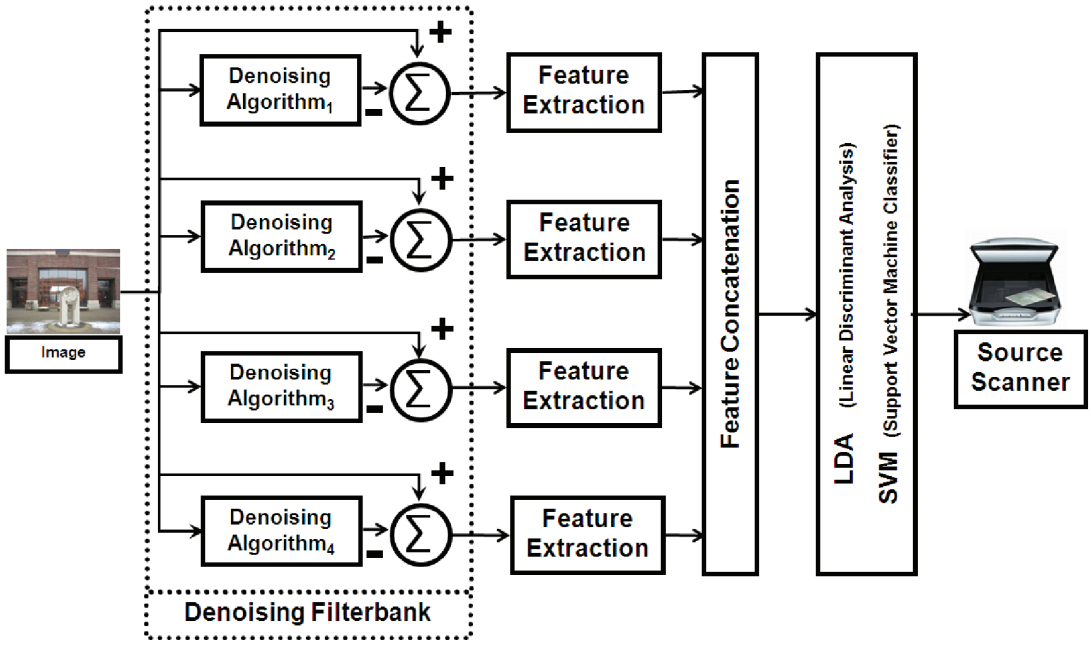


Fig. 3.4. Block Diagram of Statistical Features Based Scanner Identification Method.

pixels. This block size is arbitrarily chosen to provide for statistical significance of features used for classification, reasonable processing time and memory usage. For each of the seven scanners we have 200 sub-images from each column of the sliced images corresponding to that scanner. Figure 3.6 shows a sample of the images used in this study. As shown in Figure 3.5 the image blocks such as $B0$ and $B5$ from the same column will be scanned by the same sensor elements and can therefore be treated as originating from the same source. Unless stated otherwise, for experiments on native resolution images, 50% of the sub-images are randomly chosen for training of the SVM classifier and the remaining sub-images are used for testing.

First a set of experiments are performed to investigate the scan-area independence of the proposed statistical features. In the first experiment a classifier is designed by placing the sub-images from the first two columns of a scanner into two different

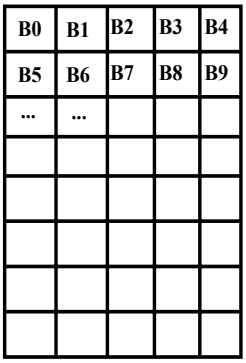


Fig. 3.5. Scanned Images Sliced into Sub-images.

Fig. 3.6. Sample Images Used for Source Scanner Identification.

classes. For example, image blocks such as $B0$ and $B5$ are in one class and image blocks $B1$ and $B6$ are in another class (Figure 3.5). Table 3.8 shows the confusion matrix for training and testing on 14 different classes, treating sub-images coming from two columns of the same scanner as two different classes. Sub-images used for generating this confusion matrix were stored in TIF format. In this table the class S_j^c denotes sub-images from the c^{th} column of the j^{th} scanner. A similar classifier for sub-images stored in JPEG format at quality factor 70 has the confusion matrix shown in Table 3.9. The results in these tables suggest that the proposed features for

sub-images from different columns of the same scanner differ from each other. For some scanners, this difference is enough to reliably differentiate sub-images from the two columns. Different columns from several scanners such as S_1, S_2, and S_7 have classification accuracies of only 75% for TIFF images, and the overall classification accuracy for JPEG images is even lower for all the scanners. This indicates that these features fall into overlapping clusters. Possible reasons for the poor classification accuracies are that both classes contain noise caused by similar mechanical fluctuations and post-processing algorithms.

Another experiment to investigate the scan-area independence of the proposed statistical features is designed to show scan-area independence by training the classifier on sub-images from the first column of the scanned images and testing on sub-images from the second column of the scanned images. Table 3.10 shows the confusion matrix for this classifier which has an average classification accuracy of 95%. A similar experiment, designed by training the classifier on sub-images from second column of the scanned images and testing on sub-images from the first column of the scanned images, has an average classification accuracy of 92%. Similar classifiers designed for images saved in JPEG format (Q=70) have classification accuracies close to 95% except for scanners S_2 and S_3 which are of the same make and model. These results indicate that even though the features from sub-images of different columns are somewhat differentiable, features from different columns of the same scanner are clustered closer to one another than to those of other scanners. Therefore for the purpose of source scanner identification, the proposed feature set can be assumed to be independent of scan-area.

For scanning at native resolutions of the scanners, the following observations may be drawn from the results of the above experiments:

- The proposed features for images scanned from different locations on the same scanner bed fall into over-lapping or non-overlapping clusters which are much closer to each other than to clusters corresponding to features for sub-images from other scanners.

Table 3.8
Using Statistical Features (treating TIFF sub-images from different horizontal locations as separate classes).

		Predicted													
		S_1^1	S_1^2	S_2^1	S_2^2	S_3^1	S_3^2	S_4^1	S_4^2	S_6^1	S_6^2	S_7^1	S_7^2	S_9^1	S_9^2
Actual	S_1^1	74.0	26.0	0	0	0	0	0	0	0	0	0	0	0	0
	S_1^2	30.0	70.0	0	0	0	0	0	0	0	0	0	0	0	0
	S_2^1	0	0	74.6	22.4	2.6	0.4	0	0	0	0	0	0	0	0
	S_2^2	0	0	20.1	76.0	1.9	2.0	0	0	0	0	0	0	0	0
	S_3^1	0	0	0.7	0.8	78.1	20.4	0	0	0	0	0	0	0	0
	S_3^2	0	0	0.0	1.9	18.6	79.4	0	0	0	0	0	0	0	0
	S_4^1	0	0	0	0	0	0	100	0	0	0	0	0	0	0
	S_4^2	0	0	0	0	0	0	0	100	0	0	0	0	0	0
	S_6^1	0	0	0	0	0	0	0	0	98.7	1.0	0	0	0.1	0.2
	S_6^2	0	0	0	0	0	0	0	0	0.7	98.8	0.3	0	0.0	0.2
	S_7^1	0	0	0	0	0	0	0	0	0	0	76.7	23.3	0	0
	S_7^2	0	0	0	0	0	0	0	0	0	0	25.1	74.9	0	0
	S_9^1	0	0	0	0	0	0	0	0	0.3	0.1	0	0	81.9	17.7
	S_9^2	0	0	0	0	0	0	0	0	0.4	1.3	0	0	15.7	82.6

Table 3.9
Using Statistical Features (treating JPEG (Q=70) sub-images from different horizontal locations as separate classes).

		Predicted													
		S_1^1	S_1^2	S_2^1	S_2^2	S_3^1	S_3^2	S_4^1	S_4^2	S_6^1	S_6^2	S_7^1	S_7^2	S_9^1	S_9^2
Actual	S_1^1	88.8	10.3	0	0.3	0.1	0.0	0.1	0	0	0	0.1	0.3	0	0.0
	S_1^2	22.8	76.1	0	0	0.0	0.3	0	0	0	0	0.0	0.7	0	0
	S_2^1	0	0	64.8	24.9	5.5	3.6	0.0	0.1	0.2	0	0.8	0	0.1	0
	S_2^2	0.3	0	20.2	57.9	7.4	12.2	0.5	0	0.2	0	0.3	0.5	0.5	0
	S_3^1	0.7	0.1	2.2	7.2	60.0	26.5	0	0.3	1.3	0.5	0.0	0	1.0	0.2
	S_3^2	0.4	0.0	4.9	11.2	18.7	60.1	0.0	0	0.2	0	1.3	0.5	1.7	0.8
	S_4^1	0.5	0	0	0	0	0	88.5	11.0	0	0	0	0	0	0
	S_4^2	0	0	0.0	0	0.1	0	8.9	91.0	0	0	0	0	0	0
	S_6^1	0	0	0	0.1	1.3	0.2	0	0	73.2	20.2	0.3	0	3.8	0.9
	S_6^2	0	0	0	0.1	0.2	0.3	0	0	31.6	64.5	0.8	0.3	1.4	0.9
	S_7^1	0	0	0.7	0.1	0.2	1.2	0	0	0.1	0	72.2	22.0	1.5	2.1
	S_7^2	1.0	0.7	0	0.3	0.2	2.0	0	0	0	0	20.0	71.9	0.3	3.4
	S_9^1	0	0	0.0	0.1	0.7	0.9	0.0	0	3.5	1.9	1.0	0.5	72.3	19.1
	S_9^2	0	0	0	0	0.3	1.2	0	0	2.0	1.7	1.8	2.1	23.0	67.9

Table 3.10

Using Statistical Features: Native Resolution TIFF Sub-images, Trained on Sub-images from Column-1 and Tested on Sub-images from Column-2.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	100	0	0	0	0	0	0
	S_2	0	89.6	6.6	0	3.3	0	0.5
	S_3	0	6.5	91	0	2	0	0.5
	S_4	0	0.5	0	93.5	6	0	0
	S_6	0	0	0	0	90.4	0.4	9.2
	S_7	0	0	0	0	0	100	0
	S_9	0	0	0	0	1	0	99

- Features for images scanned from different scanners fall into separate clusters.
- For some scanners it may be possible to distinguish between images scanned from different locations on scanner bed. In all cases, features for images scanned from the same scanner (independent of the scanning location) lie much close to each other than to features for images scanned from a different scanner.
- With the degradation in image quality due to heavy JPEG compression, separation between scanners of the same make and model decreases and the proposed features may be able to identify only the make and model of the source scanner and not the unique scanner.

These experiments show the scan-area independence of the proposed scheme. In the following experiments on native resolution images, sub-images from the first two columns of the sliced images are placed into a single class corresponding to that

scanner. This results in 400 sub-images for each of the seven scanners. Sub-images from the first two columns of the j^{th} scanner are denoted by class S_j.

3.4.2 Native Resolution Images

Native Resolution TIFF Images

Table 3.11 shows the confusion matrix corresponding to source scanner identification among seven scanners using the proposed scheme. Using 200 randomly chosen sub-images for training and the remaining 200 for testing, 100% classification is achived over seven scanners. The final decision about the source scanner of a native resolution image is taken by majority voting over the decisions corresponding to the individual sub-images. The underlying sub-image classification accuracies are less than 100% due to the fact that several sub-images may contain only saturated regions (completely black or white)s of the image in which sensor noise is not detectable.

Table 3.11
Using Statistical Features: Native Resolution, TIFF Sub-images.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	100	0	0	0	0	0	0
	S_2	0	96.6	3.4	0	0	0	0
	S_3	0	0.6	99.4	0	0	0	0
	S_4	0	0	0	100	0	0	0
	S_6	0	0	0	0	99.6	0.2	0.2
	S_7	0	0	0	0	0	100	0
	S_9	0	0	0	0	0.8	0	99.2

To compare the performance of the proposed scheme with other existing feature vector based forensic classification schemes, the Image Quality Measures (IQM)

based source camera identification method [18] and the source scanner identification method proposed by Gou et al. [26] are implemented. In [18] features such as IQM and wavelet-based features are used. In our implementation of the IQM based classifier, a 28-dimensional feature vector is extracted from each input image and LDA is performed to reduce the dimensionality of the feature space to ten. A SVM classifier using a RBF kernel is used for classification. Gou et al.'s method for source scanner identification uses three sets of features extracted from each scanned image. This method is aimed at classifying images depending upon the scanner model that generated it and not the exact scanner. The first set of features includes the mean and standard deviation of the log-absolute transformed noise estimated using five different denoising filters. The denoising filters used in this scheme are: 1) linear filtering with an averaging filter (3×3 kernel), 2) linear filtering with a Gaussian filter (3×3 kernel), 3) median filtering (3×3 kernel), and 4) Wiener adaptive image denoising with kernel sizes 3×3 and 5×5. This gives a total of 30 features from the image noise. The second set of features are based on the observation that the high-frequency wavelet coefficients of the scanned images approach a Gaussian distribution and that different scanner models fit the Gaussian model differently. The absolute value of the area under the difference of the Gaussian curve and the histogram of the high-frequency wavelet coefficients of the scanned images makes up the second set of features. The smooth regions of the scanned images may be contaminated by noise and result in non-trivial error in the neighborhood prediction. The difference in prediction error will capture the variation of scanning noise among different scanner models. The third set of features includes the mean and standard deviation of the prediction errors in smooth regions. This gives a 60-dimensional feature vector for each image. In [26] principle component analysis (PCA) is applied to reduce the dimensionality of the feature space to 25-dimensions. In our implementation of Gou et al.'s scheme we perform LDA on the 60-dimensional feature space to reduce the dimensionality of the feature space to ten. This is to ensure that we are comparing the effectiveness of different features and not the differences between classifiers such

as PCA, LDA and SVM. From the results of our implementation of these methods, it is clear that using LDA instead of PCA improves the performance of the scheme proposed by Gou et al. [26]. Our goal is to compare the end-to-end performance of different scanner identification systems and not the individual components of each system.

The experiments described earlier in Section 3.4.1 to analyze the scan-area independence were also conducted for the two existing schemes. They show similar results that indicate scan-area independence.

The confusion matrix for classifying sub-images scanned from seven different scanners at their respective native resolutions using the IQM based scheme is shown in Table 3.12. The IQM based scheme has an average sub-image classification accuracy of 89.5%. Table 3.13 shows the confusion matrix for Gou et al.'s scheme. This scheme has an average sub-image classification accuracy of 95.2%. These classification accuracies indicate that the noise based features may be better than IQM based features for source scanner identification.

Table 3.12
Using IQM: Native Resolution, TIFF Sub-images.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	94.7	0.1	0.5	0.2	1.4	0	3.2
	S_2	0.0	93.6	4.6	1.0	0.5	0.0	0.3
	S_3	1.7	6.6	86.6	1.1	2.0	0.1	2.0
	S_4	0.1	2.0	1.0	91.0	0.4	5.3	0.3
	S_6	0.7	0.9	1.8	0.3	84.8	0.6	10.9
	S_7	0.3	0	0.0	2.9	0.5	93.9	2.4
	S_9	3.2	0.1	2.4	0.2	10.1	2.1	82.0

Table 3.13
Gou. et al.'s Scheme: Native Resolution, TIFF Sub-images.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	99.6	0	0	0	0	0	0.4
	S_2	0.0	86.8	12.4	0.3	0.4	0	0
	S_3	0.0	15.3	84.4	0.0	0.0	0	0.2
	S_4	0.1	0.0	0.1	99.7	0.0	0.1	0.0
	S_6	0.1	0.0	0.1	0.0	98.5	0	1.3
	S_7	0	0.0	0	0.1	0.5	99.3	0.1
	S_9	0.2	0.0	0.2	0	1.5	0	98.1

Effect of JPEG Compression

To further examine the robustness of the proposed approach, experiments are conducted on JPEG compressed images. A dedicated SVM classifier used for this experiment is trained and tested using only JPEG compressed images. All the scanned images are JPEG compressed with quality factor Q = 70 after which feature extraction is performed. The dedicated SVM classifier is trained using randomly chosen 50% of the compressed images and tested on the remaining compressed images.

Table 3.14 shows the confusion matrix for classifying sub-images from JPEG compressed images with Q = 70. An average sub-image classification accuracy of 92% is achieved in this case. Tables 3.15 and 3.16 show the confusion matrices for the IQM based scheme and Gou et al.'s scheme respectively. For this experiment the IQM based scheme has an average sub-image classification accuracy of 68.6% while Gou et al.'s scheme has an average sub-image classification accuracy of 80.8%. These results also show that the separation between scanners of the same make and model decreases with degradation in the noise pattern due to JPEG compression. The lower

decline in performance due to JPEG compression for the proposed scheme suggests that the proposed features are more robust to JPEG compression.

Table 3.14

Using Statistical Features: Native Resolution Sub-images, JPEG Compressed (Q=70), Dedicated Classifier.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	99.2	0.2	0.2	0.0	0	0.4	0
	S_2	0.1	83.3	15.6	0.3	0.0	0.5	0.1
	S_3	0.7	10.1	85.3	0.2	1.4	0.8	1.6
	S_4	0.5	0	0	99.5	0	0	0
	S_6	0	0.3	0.9	0	93.9	0.6	4.2
	S_7	1.1	0.7	1.1	0	0.0	92.9	4.2
	S_9	0	0	1.8	0	5.2	3.0	90.0

3.4.3 Non-native Resolution Images

In the next few experiments, the effectiveness of the proposed scheme is shown for heavily sub-sampled (200 DPI) images. These experiments have a broad practical impact since most scanned images are at lower non-native resolutions due to limitations on storage space and transmission speed. The scheme proposed here has good performance on 200 DPI images (which corresponds to scaling by 17% to 4% for native resolutions of 1200 DPI to 4800 DPI, respectively).

Table 3.15

Using IQM: Native Resolution Sub-images, JPEG Compressed (Q=70), Dedicated Classifier.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	91.0	0.8	1.8	4.2	0.1	1.4	0.7
	S_2	0.9	50.7	33.8	1.4	1.5	4.1	7.5
	S_3	2.3	32.7	48.0	2.0	2.7	5.7	6.7
	S_4	8.5	3.8	4.4	78.7	0.7	1.2	2.7
	S_6	1.1	2.1	1.0	0.7	83.1	1.6	10.5
	S_7	2.0	5.9	6.8	1.6	2.6	68.9	12.2
	S_9	2.5	8.7	4.6	3.3	10.6	10.5	59.9

Table 3.16

Gou et al.'s Scheme: Native Resolution Sub-images, JPEG Compressed (Q=70), Dedicated Classifier.

		Predicted						
		S_1	S_2	S_3	S_4	S_6	S_7	S_9
Actual	S_1	94.7	1.2	0.7	1.7	0.2	0.8	0.6
	S_2	2.7	61.3	30.8	1.1	0.4	2.1	1.5
	S_3	1.6	29.5	59.3	2.1	0.7	4.1	2.7
	S_4	1.5	1.2	2.1	91.8	0.6	1.1	1.7
	S_6	0.6	0.5	1.4	0.3	92.7	0.4	4.1
	S_7	5.1	1.9	2.7	0.3	0.3	85.9	3.7
	S_9	2.8	2.1	3.8	1.6	6.4	3.3	80.0

Non-native Resolution TIFF Images

For performing the experiments on lower resolution images, 108 images are scanned at 200 DPI using each of the eleven scanners shown in Table 3.1. Each scanned image is saved as an uncompressed TIFF image. It is not necessary to divide these low resolution images into smaller blocks because they are small enough to process in a reasonable amount of time. Therefore, feature extraction and classification is performed over an entire image and not multiple sub-images. Unless stated otherwise, for each experiment on 200 DPI images, 80 randomly selected images from each class are used for training and the remaining images are used for testing.

Figure 3.7 shows a scatter plot of the first two features obtained after application of LDA on the 204 dimensional feature vectors corresponding to the uncompressed TIFF images from six scanner classes. These six scanner classes S_1, $(S_2 + S_3)$, S_4, S_5, S_{10} and S_{11} have the largest separation in this two dimensional feature space. This scatter plot gives an indication of the high accuracy of the proposed scheme since even in the two dimensional feature space six scanner classes can be easily separated. In this two dimensional feature space the features for images from S_2 and S_3 are non-separable, however, together they form one cluster which is separate from all other classes. This is due to S_2 and S_3 both being of the same make and model.

Degradation in the characteristics of sensor noise due to heavy down-sampling prevents successful separation of images scanned from the two scanners of exact same make and model as demonstrated by our initial experiments (Tables 3.17 and 3.18). In the training and testing phases of these experiments the images from scanners S_2 and S_3 are treated as images coming from two different sources. It appears that the low resolution images from scanners of the same make and model are not clearly separable using the proposed features. As shown in Table 3.17, only 90% of the 200 DPI TIFF images from scanners S_2 and S_3 are classified correctly. This separation further decreases to 75% with JPEG compression (Table 3.18). Image classification accuracies for all other scanners are close to 100% for TIFF images and 90% for

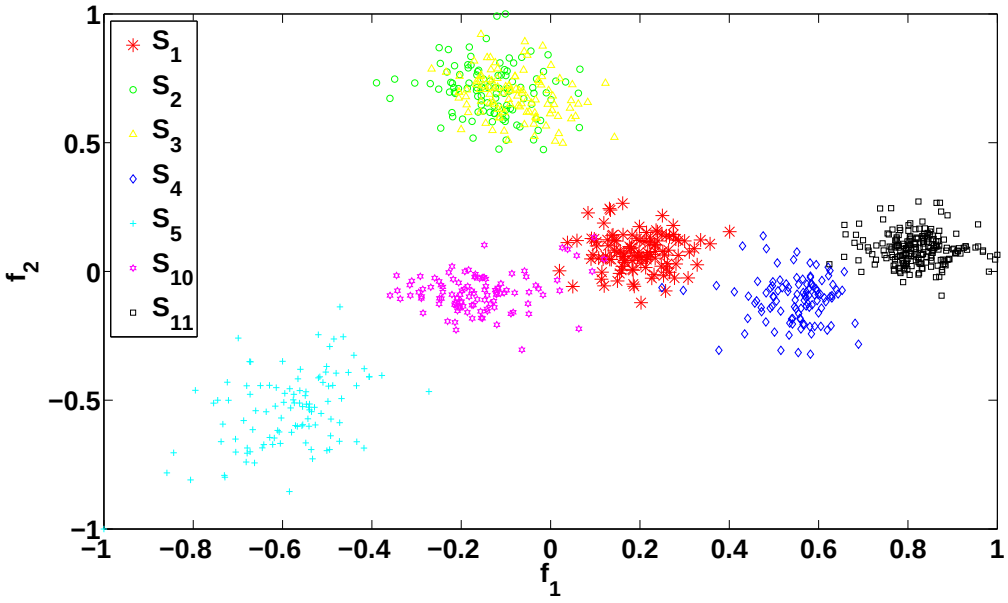


Fig. 3.7. Scatter Plot of First Two Features of the Proposed Scheme (for six classes having the best separation in 2D projected feature space).

JPEG compressed images. Therefore, the following experiments performed on images scanned at 200 DPI are focused on classifying images based on the scanner make and model and treat scanners S_2 and S_3 as a single class. The scatter plot shown in Figure 3.7 also supports a similar conclusion.

Table 3.19 shows the confusion matrix for classifying images from eleven scanners of ten different make and models using the proposed scheme. Note that scanners S_2 and S_3 of the same make and model are treated as one class. The proposed algorithm has an average classification accuracy of 99.9% among ten scanner models. Table 3.20 shows the corresponding confusion matrix for eleven scanners using the IQM based scheme, which has an average classification accuracy of 88.4%. Table 3.21 shows the confusion matrix for classifying TIFF images from eleven scanners using the scheme proposed by Gou et al. Gou et al.'s scheme has an average classification accuracy

Table 3.17

Using Statistical Features: 200 DPI TIFF Images, Treating S_2 and S_3 as Distinct Classes (training set: 80 images from each class).

		Predicted										
		S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	100	0	0	0	0	0	0	0	0	0	0
	S_2	0	93.2	6.8	0	0	0	0	0	0	0	0
	S_3	0	10.1	89.9	0	0	0	0	0	0	0	0
	S_4	0	0	0	100	0	0	0	0	0	0	0
	S_5	0	0	0	0	100	0	0	0	0	0	0
	S_6	0	0	0	0	0	100	0	0	0	0	0
	S_7	0	0	0	0	0	0	100	0	0	0	0
	S_8	0	0	0	0	0	0	0	100	0	0	0
	S_9	0	0	0	0	0	0	0	0	100	0	0
	S_{10}	0.3	0	0	0	0	0	0	0.2	0	99.5	0
	S_{11}	0	0	0	0	0	0	0	0	0	0	100

of 96.6%. The proposed scheme based on the statistical features of the sensor noise performs better for source scanner identification than the IQM based scheme and Gou et al.'s scheme.

At this point it is interesting to compare the classification accuracies for classifying native resolution images with those for classifying non-native resolution images (Table 3.11 vs. Table 3.19). The reason for the differences in classification performances between native and non-native resolution images (Table 3.11 vs. Table 3.19) lies in the way experiments are designed and the fact that the pattern noise is not detectable in saturated (completely black or white) regions of the image. For the experiments on native resolution images the original scanned image is first divided into smaller blocks of size 1024×768, then a feature vector is generated for each block and the classification decisions are taken for each block separately. These block wise decisions have false classifications for the sub-images corresponding to saturated (completely

Table 3.18

Using Statistical Features: 200 DPI JPEG Images (Q=90,80,70), Treating S_2 and S_3 as Distinct Classes , (training set: 80 images from each class consisting of all three quality factors).

		Predicted										
		S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	88.6	1.5	3.2	0.0	0.8	0	1.7	1.9	1.1	1.0	0.3
	S_2	0.6	75.6	19.8	0.0	0	0	0	1.9	1.5	0.2	0.4
	S_3	1.2	16.5	77.3	0.0	0.2	0	0.3	2.4	1.7	0.3	0.1
	S_4	1.3	0.4	0.4	97.2	0.4	0	0.2	0	0	0.2	0
	S_5	1.9	0.2	0.2	0	93.5	1.0	0.9	0.7	0.6	0.9	0
	S_6	0	0.1	0	0	0.8	97.3	0.4	0.2	0.9	0.4	0
	S_7	0.6	0.3	0.6	0	0.3	1.3	87.1	7.7	1.8	0.2	0.2
	S_8	0.3	1.7	3.3	0	0.1	0	6.7	86.0	1.4	0.5	0.1
	S_9	2.7	0.7	1.3	0.1	0.6	0.7	1.5	2.5	89.7	0.3	0
	S_{10}	3.0	0.1	0.7	0.3	2.1	0.8	0.5	1.6	0.9	90.0	0
	S_{11}	0.0	0	0	0	0.1	0.1	0.3	0	0	0	99.5

black or white) regions [37]. The final decision about the source scanner of a native resolution image is determined by majority voting over the decisions corresponding to the individual sub-images. Thus even with a classification accuracy close to 95% for sub-images (Table 3.11), the final classification accuracy for the complete native resolution image remains 100%. For non-native resolution images, feature extraction and classification is performed over an entire image and not multiple sub-images. This avoids misclassification due to saturation of pixel values unless the entire image is black or white. Hence, the proposed scheme gives 100% classification accuracy for classifying the native resolution images as well as the non-native resolution images.

To further check the robustness of the proposed scheme for scanner model identification, a SVM classifier is trained without images from scanner S_3 and tested on only images from the scanner S_3. Table 3.22 shows the confusion matrix for this case,

Table 3.19

Using Statistical Features: 200 DPI TIFF Images, (training set: 80 images from each class).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	100	0	0	0	0	0	0	0	0	0
	S_2+S_3	0	100	0	0	0	0	0	0	0	0
	S_4	0	0	100	0	0	0	0	0	0	0
	S_5	0	0	0	100	0	0	0	0	0	0
	S_6	0	0	0	0	100	0	0	0	0	0
	S_7	0	0	0	0	0	100	0	0	0	0
	S_8	0	0	0	0	0	0	100	0	0	0
	S_9	0	0	0	0	0	0	0	100	0	0
	S_{10}	0.3	0	0	0	0	0	0.3	0	99.4	0
	S_{11}	0	0	0	0	0	0	0	0	0	100

which has a classification accuracy of 95%. A similar experiment designed by training the classifier without images from scanner S_2 and testing only on images from S_2, gives a classification accuracy of 97%. These results imply that even in the absence of the training data from a particular scanner, the proposed scheme can identify the scanner model as long as training data from another scanner of the same make and model is available.

Another aspect of robustness is independence from scanning location. In other words, even when the image is placed at a random unknown location on the scanner bed, source scanner identification should still be possible. The images used in all the earlier experiments were scanned from the "default" scanning location (generally marked at the top right corner) of the scanner. For this experiment another 108 images are scanned from scanner S_{11}, with their location on the scanner's bed slightly translated horizontally and vertically between each scan. A SVM classifier is trained

Table 3.20
Using IQM: 200 DPI TIFF Images, (training set: 80 images from each class).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	70.9	4.3	1.2	3.4	0	1.1	0.6	18.1	0.5	0
	S_2+S_3	3.3	83.8	0	0	0	5.1	4.7	3.1	0	0
	S_4	0.4	0.5	97.8	1.3	0	0	0	0	0	0
	S_5	1.9	0	0	94.8	0	0	0	1.6	1.7	0
	S_6	0	0	0	0	100	0	0	0	0	0
	S_7	0.8	10.9	0	0	0	80.8	3.4	4.0	0	0
	S_8	0.7	10.9	0	0	0	5.1	82.9	0.4	0	0
	S_9	7.3	5.3	0	4.8	0	4.0	0.4	78.3	0	0
	S_{10}	0	0.8	0	2.4	0	0	0	1.0	95.8	0
	S_{11}	0	0	0	0.3	0	0	0	0	0	99.7

using the images scanned from the "default" location and tested using images from the "random" locations only. Table 3.23 shows the classification results for scanner S_{11} which has a classification accuracy of 100% for "randomly" placed images. This suggests that the proposed scheme for scanner model identification is independent of the scanning location.

3.4.4 Effect of Post Processing

The following experiments are aimed at investigating the influence of post-processing operations such as JPEG compression, contrast stretching and brightness enhancement, on source scanner identification. To test whether sensor noise survives these operations, two types of classifiers are used. First is a dedicated classifier which is trained and tested only on a particular class of post-processed images. Second is a general classifier which is trained on both the original and post-processed images

Table 3.21

Gou et al.'s Scheme: 200 DPI TIFF Images, (training set: 80 images from each class).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	96.0	1.5	0	0.6	0	0	0	1.9	0	0
	S_2+S_3	0.5	98.4	0.6	0	0	0.6	0	0	0	0
	S_4	0	0	100	0	0	0	0	0	0	0
	S_5	0.2	0	0	96.5	0	0	0	0.5	2.8	0
	S_6	0	0	0	0	99.2	0.2	0.6	0	0	0
	S_7	0	2.0	0	0	0.2	94.4	3.3	0	0	0
	S_8	0	0.6	0	0	0	3.8	94.8	0	0	0.7
	S_9	7.7	0.5	0	1.3	0	0.5	0	90.0	0	0
	S_{10}	0.2	0	0	3.0	0	0	0	0	96.8	0
	S_{11}	0	0	0	0	0	0	0	0	0	100

Table 3.22

Using Statistical Features: 200 DPI TIFF Images, (training set: 80 images from each class, no image from S_3; testing set: 108 images from S_3).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_3	0	95	0	1	0	0	0	1	3	0

and tested only on the post-processed images. Unless stated otherwise, in these experiments 80 randomly selected images from each scanner class are used for training and the remaining images are used for testing. Since the proposed features are based on sensor noise, if a post-processing or malicious attack involves subtraction of the noise from the original image or addition of a spurious noise pattern, classification

Table 3.23

Using Statistical Features: Effect of Changing Scanning Location, 200 DPI TIFF Images, (training set: 80 images from each class, testing set: 108 images from random locations on S_{11}).

		Predicted									
		S_1	$S_2 + S_3$	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_{11}	0	0	0	0	0	0	0	0	0	100

accuracies are expected to decrease, similar to the performance decline noticed with JPEG compression.

Effect of JPEG Compression

To investigate the robustness of the proposed scheme under JPEG compression, TIFF images from all the scanners are compressed at three different quality factors Q = 90, 80 and 70. This gives a total of $11 \times 108 \times 3 = 3564$ JPEG images. To see the effect of JPEG compression on the proposed statistical features, one dedicated classifier is trained for each quality factor. For designing these dedicated classifiers, 80 images are randomly chosen (compressed at that quality factor) from each scanner model for training. The remaining images at that quality factor are used for testing. This training and testing is repeated multiple times to generate the final confusion matrices. Similar dedicated classifiers are designed for the IQM based scheme and for the scheme proposed by Gou et al.

The bar graph shown in Figure 3.8 shows the comparative performance of these three methods for source scanner model identification using images stored in uncompressed TIFF and JPEG format at different quality factors. The average classification accuracies over ten scanner models for the proposed scheme are 97.4%, 95.7% and 93.3% for dedicated classifiers at quality factors 90, 80 and 70 respectively. Thus, the

proposed features survive low quality factor JPEG compression. Even though there is a slight decay in the performance with decrease in JPEG quality factor, the proposed scheme maintains an average classification accuracy of 93.3% at quality factor 70. Furthermore, as is clear from the bar graph in Figure 3.8, the proposed features perform consistently better than the other two schemes.

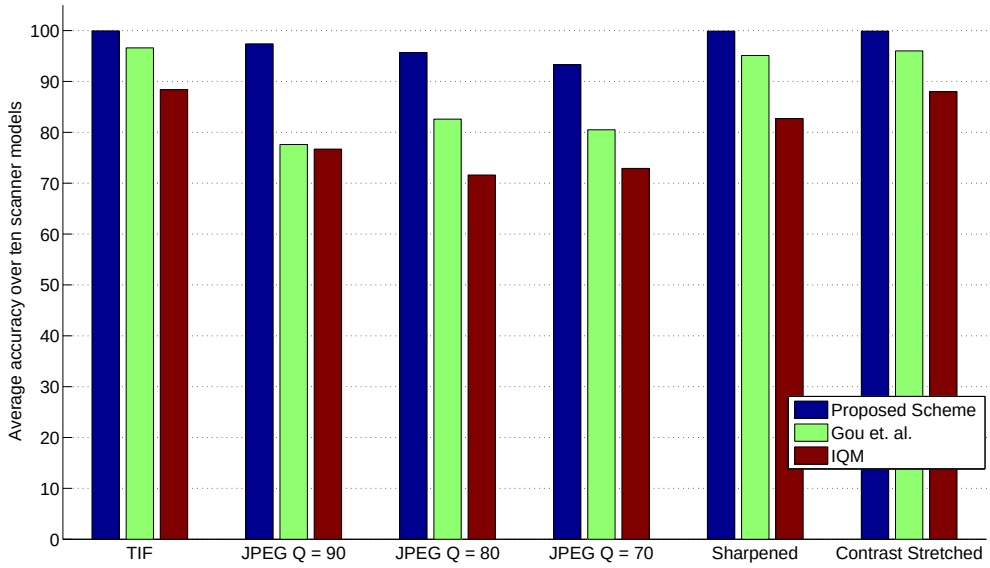


Fig. 3.8. Comparative Performance of Dedicated Classifiers for Different Schemes.

To use a dedicated classifier on post-processed images, we need to know the particular post-processing that was applied to the image. In some cases, this a priori information is available or can be obtained by using other forensic methods. For example, it may be possible o obtain the JPEG quality factor through analysis of quantization tables embedded in the JPEG image. But in general, an image of unknown origin is provided for forensic examination without reliable knowledge of the post-processing operations applied to it. Thus, there is need for a general classifier which does not need to know the JPEG quality factors of training and testing images.

To design a robust general classifier, the JPEG images compressed at three quality factors are grouped together and 80 randomly chosen images from each scanner class are used for training the classifier. The remaining $3564 - 10 \times 80 = 2764$ images are used for testing of the classifier. A similar general classifier is also designed for the IQM based scheme and the method proposed by Gou et al. Table 3.24 shows the confusion matrix for the general classifier for the proposed scheme, which has an average classification accuracy of 92.3%. Table 3.25 shows the confusion matrix for the corresponding general classifier for the IQM based scheme, which has an average classification accuracy of 75%. Table 3.26 shows the confusion matrix for the corresponding general classifier for the scheme proposed by Gou et al., which has an average classification accuracy of 57.7%. The previous schemes in their present form can be used when the JPEG quality factor of the test image is known or can be accurately estimated. However, in the general scenario considered here when the JPEG quality factor is unknown, they do not perform well. The proposed scheme gives high classification accuracy even without knowledge of the JPEG quality factors of training or testing images.

Effect of Image Sharpening and Contrast Stretching

To investigate the robustness of the proposed scheme on images that have undergone image sharpening and contrast stretching, TIFF images from all the scanners are independently sharpened and contrast stretched. A sharpening algorithm based on weighted median filtering is used (with sharpening parameter $\tau = 0.2$) [69]. The contrast stretching curve used here is depicted in Figure 3.9 and a threshold T = 20 is used. The set of images used for these experiments consists of 108 TIFF images from each of the 11 scanners, and their contrast stretched and sharpened versions, for a total of $11 \times 108 \times 3 = 3564$ images.

Figure 3.8 shows the comparative performance of dedicated classifiers for sharpened and contrast stretched images. These classifiers are trained and tested only on

Table 3.24

Using Statistical Features: General Classifier, 200 DPI JPEG (Q=90,80,70) Images, (training set: 80 images from each class consisting of all three quality factors; remaining images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	89.4	5.4	0.1	0.6	0	1.5	1.3	0.7	0.6	0.3
	S_2+S_3	0.6	96.9	0.0	0.1	0.0	0.2	1.1	0.9	0.1	0.1
	S_4	1.0	1.0	97.4	0.1	0	0.1	0	0.1	0.3	0
	S_5	2.3	0.8	0.1	92.9	1.0	0.7	0.8	0.5	1.0	0
	S_6	0.0	0.0	0	0.9	97.2	0.5	0.2	0.7	0.4	0
	S_7	1.2	1.4	0.0	0.4	1.5	85.2	8.2	1.5	0.3	0.2
	S_8	0.5	6.5	0.1	0.3	0.0	5.6	85.5	1.1	0.4	0.0
	S_9	2.5	1.9	0.1	0.7	0.7	1.5	2.7	89.5	0.4	0
	S_{10}	3.6	1.0	0.5	1.9	0.9	0.5	1.7	0.8	89.2	0
	S_{11}	0.0	0	0	0.0	0.1	0.3	0	0	0	99.5

images that have undergone that particular post-processing. This shows that only image sharpening has a significant effect on the performance of the IQM based scheme and that the sensor noise based schemes are unaffected by image sharpening and contrast stretching if the type of post-processing is known.

For building a general classifier, all the TIFF images (original and post-processed) are grouped together and 80 randomly chosen images from each scanner class are used for training of the classifier. The remaining sharpened and contrast stretched images are used for testing of the classifier, i.e. only post-processed images are used for testing. This general classifier is also designed for the IQM based scheme and the method proposed by Gou et al.Table 3.27 shows the confusion matrix for the general classifier for the proposed scheme which has an average classification accuracy of 99.8%. Table 3.28 shows the confusion matrix for the corresponding general classifier

Table 3.25

Using IQM: General Classifier, 200 DPI JPEG Images(Q=90,80,70), (training set: 80 images from each class consisting of all three quality factors; remaining images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	67.9	6.2	0.1	5.1	0.8	1.9	4.8	11.2	1.5	0.6
	S_2+S_3	1.4	79.8	0.1	1.0	0.5	5.9	7.1	1.8	0.3	2.2
	S_4	0.4	2.2	94.0	0.9	0.4	0.1	0	1.5	0.5	0
	S_5	5.5	6.9	0.4	78.1	1.3	0.9	1.4	1.6	3.9	0.1
	S_6	1.6	2.1	0.2	0.7	85.0	2.0	0.5	6.6	0.9	0.5
	S_7	1.5	15.7	0.1	0.8	0.7	57.4	8.3	7.9	0.1	7.6
	S_8	5.4	19.8	0	0.3	0.8	9.6	53.9	8.1	0	2.1
	S_9	8.5	9.5	1.0	0.8	3.8	7.3	9.3	57.2	0.9	1.6
	S_{10}	1.3	1.4	0.2	3.6	3.0	1.2	1.6	2.7	84.7	0.4
	S_{11}	0.0	0.7	0	0.0	0.2	2.2	1.1	0.6	0.3	94.9

for the IQM based scheme, which has an average classification accuracy of 79.7%. Table 3.29 shows the confusion matrix for the corresponding general classifier for the scheme proposed by Gou et al., which has an average classification accuracy of 95.4%. The average classification accuracy of methods based on sensor noise is not affected by image sharpening and contrast stretching while the IQM based scheme shows a significant drop in performance. The proposed scheme gives high classification accuracy, even on images that have undergone image sharpening and contrast stretching, without any knowledge of the post-processing performed on the training or testing images.

Table 3.26

Gou et al.'s Scheme: General Classifier, 200 DPI JPEG Images (Q=90,80,70), (training set: 80 images from each class consisting of all three quality factors; remaining images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	43.5	10.7	4.9	3.9	10.0	4.0	3.9	9.6	7.4	2.3
	S_2+S_3	2.5	74.8	1.5	4.2	0.8	4.3	2.5	3.7	1.9	3.9
	S_4	4.3	4.1	68.6	5.9	3.4	2.5	1.0	6.2	3.8	0.3
	S_5	2.5	12.9	3.8	68.2	1.1	2.4	0.5	2.8	5.9	0
	S_6	8.7	1.9	1.2	1.4	70.3	2.6	2.6	5.2	3.8	2.2
	S_7	2.8	9.8	1.5	1.7	3.1	40.5	16.7	5.9	3.3	14.6
	S_8	6.3	9.2	0.8	0.2	4.5	15.2	44.7	8.2	2.5	8.4
	S_9	13.4	11.6	5.0	3.1	9.4	5.8	5.9	38.0	4.7	3.0
	S_{10}	11.7	9.0	5.5	9.3	10.0	3.2	3.8	5.1	40.3	2.1
	S_{11}	0.4	3.2	0.1	0	1.3	4.2	2.7	0.3	0.1	87.7

3.4.5 Effect of Number of Training Images

The classifier for original TIFF images and the general classifiers designed for JPEG compressed images are the most relevant for practical applications. The next series of experiments are designed to determine the effect that the number of available training images has on the average classification accuracy. The number of training images from each scanner class varies from 10 to 90. Figure 3.10 shows the effect of the number of training images on average classification accuracy for general classifiers for different training and testing sets and different schemes.

Figure 3.11 shows the effect of changing the size of training dataset on average classification accuracy when classifying native-resolution sub-images from seven scanners. High classification accuracy is achieved even with just 20 sub-images from each scanner. These classification accuracies are for classifying 1024×768 sub-images and

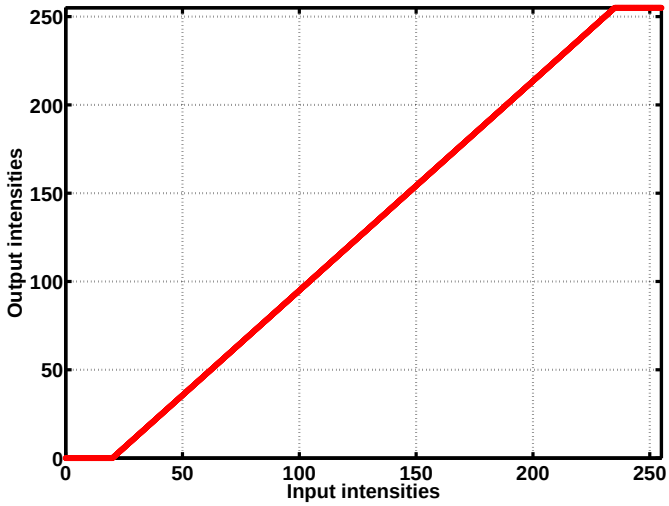


Fig. 3.9. Contrast Stretching Curve.

not the complete scanned images. Thus, even with classification accuracies close to 90% for sub-images, the final classification accuracies for classifying the complete images will remain 100%. The sub-images containing completely dark regions are generally mis-classified due to suppression of noise in dark regions. The total number of sub-images used is 1600. Similarly, varying the training size for source scanner identification among seven scanners shows that the average classification accuracy remains close to 90% for training sizes varying from 160 to 360 sub-images from each scanner.

3.4.6 Effectiveness of Different Denoising Algorithms

To investigate the source of the high accuracy achieved by our proposed scheme, the next set of experiments use the proposed noise features from each of the four denoising algorithms independently to design four separate classifiers. Average classification accuracies given by these four classifiers are compared with the average classification accuracy achieved using the denoising filter bank. For example, LDA

Table 3.27

Using Statistical Features: General Classifier, 200 DPI TIFF Images (original, sharpened, contrast stretched), Proposed Scheme, (training set: 80 images from each class consisting of all three types; remaining post-processed images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	99.6	0	0	0	0	0.1	0.0	0	0.3	0
	S_2+S_3	0	100	0	0	0	0	0	0	0	0
	S_4	0	0	99.8	0	0	0	0	0	0.2	0
	S_5	0	0	0	100	0	0	0	0	0	0
	S_6	0	0	0	0	100	0	0	0	0	0
	S_7	0	0	0	0	0	100	0	0	0	0
	S_8	0	0	0.1	0	0	0	99.7	0	0.2	0
	S_9	0	0	0	0	0	0	0	100	0	0
	S_{10}	0.6	0	0	0.1	0	0	0.2	0	99.1	0
	S_{11}	0	0	0	0	0	0	0	0	0	100

is applied on 51 features extracted using the LPA-ICI denoising algorithm [64] and ten dimensional feature vectors are obtained for each TIFF image. A dedicated classifier trained using 80 images from each scanner class gives an average classification accuracy of 97.5%, as shown by the first bar in Figure 3.12. Similar steps are applied to design dedicated classifiers using noise features from the three other denoising algorithms and for different levels of JPEG compression. With the decrease in JPEG quality factor, the average classification accuracy decreases rapidly for all four denoising algorithms, however the average classification accuracy achieved by the combined filter bank remains greater than 90% even at JPEG quality factor 70. Hence, the design of suitable noise features and use of a denoising filter bank which can capture different types of scanning noise results in the consistently high classification accuracy achieved by the proposed scheme.

Table 3.28

Using IQM: General Classifier, 200 DPI TIFF Images (original, sharpened, contrast stretched), (training set: 80 images from each class consisting of all three types; remaining post-processed images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	66.6	3.7	0.2	7.3	3.9	0.7	0.5	13.8	3.3	0.1
	S_2+S_3	3.2	84.9	1.1	2.1	1.0	2.6	1.9	2.9	0.1	0.3
	S_4	1.3	2.6	93.5	0.8	0	0	0	0.2	1.6	0
	S_5	5.4	0.6	0.5	82.7	0.2	0.4	0	0.5	9.6	0
	S_6	5.4	4.3	1.0	0.1	81.4	1.3	0.6	5.1	0.8	0
	S_7	3.5	9.7	0.0	3.1	10.2	59.7	4.4	9.1	0.4	0
	S_8	1.9	13.6	0.2	0.3	2.3	8.3	65.2	8.2	0	0
	S_9	11.8	2.3	0.5	2.2	5.5	1.8	0.9	71.5	3.4	0
	S_{10}	2.1	0.0	0.1	6.0	0	0	0	0	91.8	0
	S_{11}	0	0.7	0	0	0	0	0	0	0	99.3

3.5 Forgery Detection in Scanned Images

The statistical feature based method for source scanner identification (Chapter 3 [22,23,37]) can be extended to obtain a digital forensic tool for forgery detection in scanned images [39]. Given an image from one of the scanners in our training database, the aim is to determine the authenticity of the image and to identify the source scanner. Further, if the image is tempered by changing the image content then the algorithm should identify the manipulated regions. It is assumed that the manipulator did not have knowledge of or access to the actual source scanner and thus the changed image content is coming from images obtained from other sources.

Applicability of this method is limited to copy-paste forgeries created by copying a portion of one scanned image and pasting it into another image scanned using a different scanner. If some forgery is created by copying and pasting certain regions

Table 3.29

Gou et al.'s Scheme: General Classifier: 200 DPI TIFF Images (original, sharpened, contrast stretched) (training set: 80 images from each class consisting of all three types; remaining post-processed images for testing).

		Predicted									
		S_1	S_2+S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}
Actual	S_1	91.3	0.8	2.6	2.7	0.0	0.8	0	1.0	0.7	0
	S_2+S_3	0.6	98.6	0.2	0.0	0	0	0.0	0.4	0.0	0.1
	S_4	0	0.1	98.1	0.1	0.5	0	0.3	0.6	0.2	0
	S_5	0.8	0	0.5	94.8	0	0	0	0.1	3.8	0
	S_6	0	0	0.4	0	98.7	0.1	0.4	0.4	0	0.1
	S_7	3.2	0.7	0	0	0.5	92.6	1.8	1.0	0	0.1
	S_8	0	0	0	0	0.6	2.4	96.3	0	0	0.7
	S_9	8.4	0.4	0.1	0.1	0.4	0.9	0	89.1	0.6	0
	S_{10}	0.4	0	0.8	2.3	0	0	0	0.1	96.4	0
	S_{11}	0	0	0	0	0	0	1.3	0	0	98.7

from the images scanned using the same scanner, then proposed algorithm will fail to identify those manipulation and will instead declare these images as non-manipulated images. This is because the selected features are independent of image content and scan area and remain fixed with a particular scanner. For this class of forgeries the methods presented in [70, 71] can be used.

3.5.1 Forgery Detection Method

The proposed method detects forged regions by using image sensor pattern noise which is a unique fingerprint of the imaging sensor and was used earlier in [53] for detecting tempered regions in digital camera images. The basic idea is to divide the unknown image into smaller blocks and classify each block separately for finding

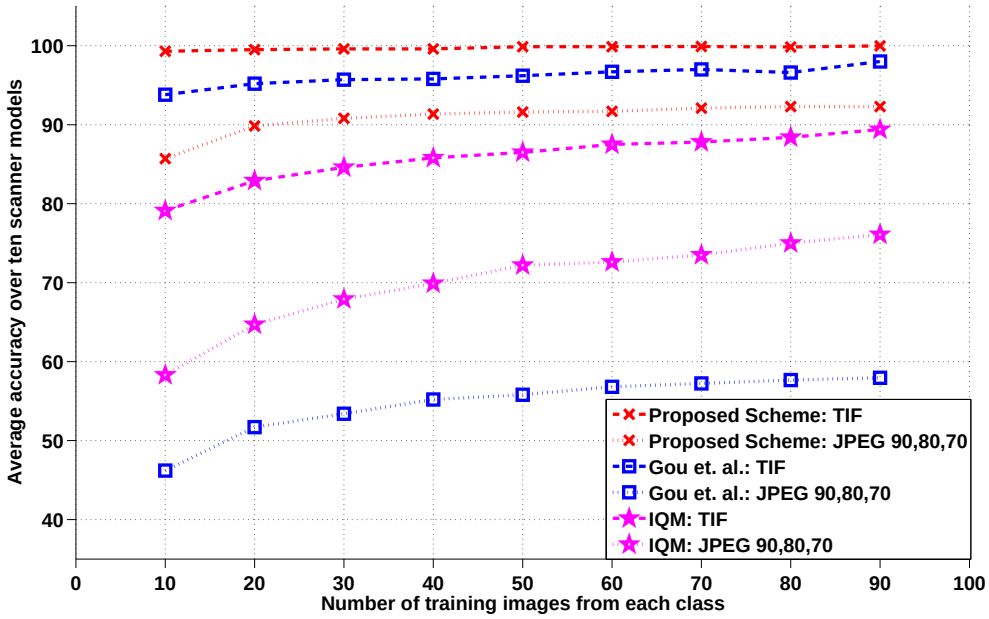


Fig. 3.10. Effect of Training Size on Average Classification Accuracy (for non-native resolution images).

out its source scanner. If all the blocks in an image are declared as coming from a single source scanner, the image is declared as an authentic image coming from that source scanner. Otherwise, different regions are coming from different sources and thus the image is a forged image. This division of the image into smaller blocks can be done either by sliding a non-overlapping window or by sliding an overlapping window. The first approach will have much lower complexity compared to the second approach, while giving a much coarser result. In second approach, feature vectors will be extracted for each pixel (except some boundary pixels depending upon the size of sliding window) of the image by using a window centered on that pixel. The sliding window dimensions impose limitations on the lower bound of the dimensions of forged regions detected. Thus, similar to [72], in the decision map obtained in second approach connected components smaller than half the window size are removed. Next

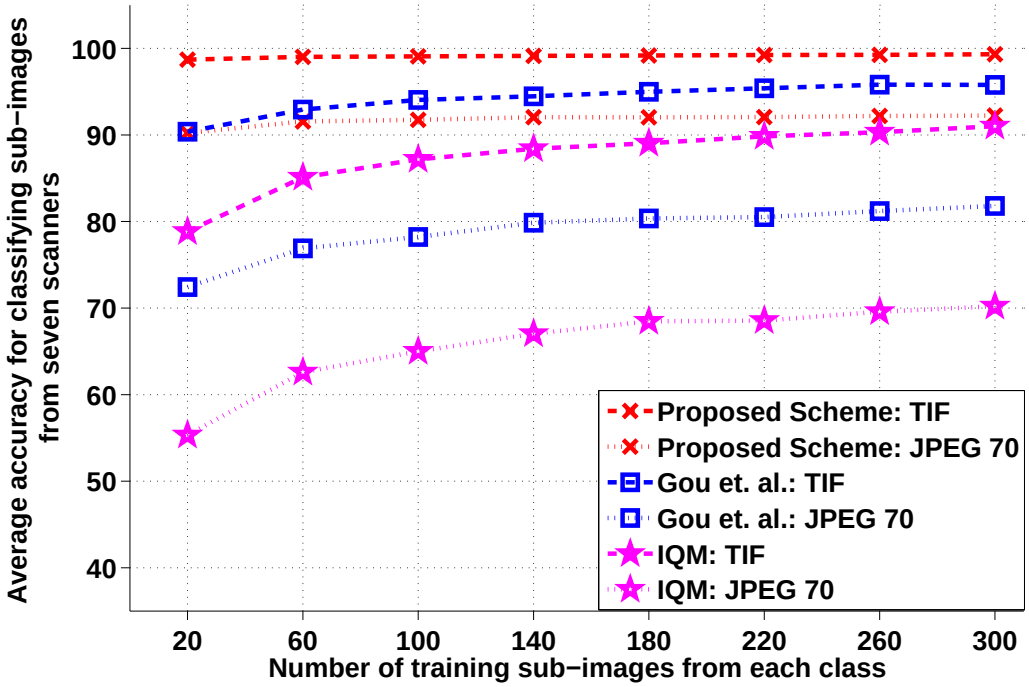


Fig. 3.11. Effect of Training Size on Average Classification Accuracy (for native resolution images).

this decision map is dilated with a small kernel to accommodate the fact that a decision about entire window is assigned only to the central pixel which may result in missing the portions of the forged boundary regions.

Statistical features of sensor noise for each instance of a sliding window are extracted and these blocks are independently classified for source scanner using a Support Vector Machine (SVM) classifier. The image is declared to be an authentic image coming from scanner S_i if all the blocks are classified as originating from scanner S_i. If the image contains regions from more than one source it is declared as forged image and the forged regions are also identified. This method is applicable whenever we have access to the scanner (or authentic images scanned using that scanner) claimed as the source of the test image.

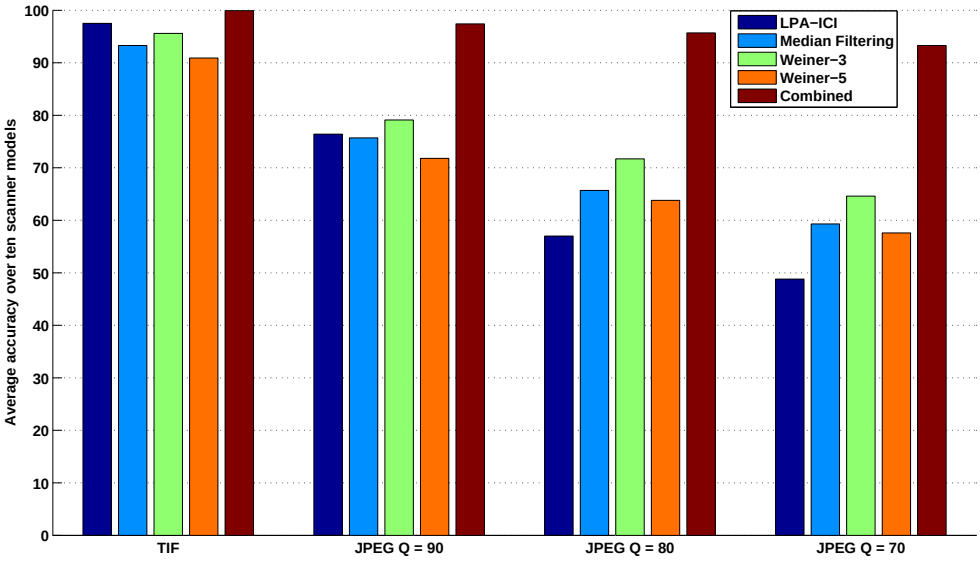


Fig. 3.12. Effectiveness of Different Denoising Algorithms Used by the Proposed Scheme.

3.5.2 Experimental Results

This section describes the details of experiments conducted to examine the efficacy of the proposed algorithm for forgery detection in scanned images. Table 3.30 shows the scanners used in our experiments. For training the classifier, approximately 25 images are scanned with each of the 5 scanners (a total of approximately 125 images) at the native resolution of the scanners. These images are then sliced into blocks of size 384×512 pixels. Thus, in total, we have approximately 2000 scanned sub-images.

A SVM classifier is trained using the feature vectors for the sub-images from authentic images of known origin. Several forged images were created by copymove within the same image and adding or covering objects using images from two different scanners. Representative forgery detection results for each type of forgeries is presented here, along with the description of the forgeries.

Table 3.30
Scanner Set Used for Evaluation of Forgery Detection Method.

	Make/Model	Type	Sensor	Native Resolution (DPI)
S_1	HP ScanJet 6300c-1	Flatbed	CCD	1200
S_2	HP ScanJet 6300c-2	Flatbed	CCD	1200
S_3	Visioneer OneTouch 7300	Flatbed	CIS	1200
S_4	Canon LiDe 25	Flatbed	CIS	1200
S_5	OpticSlim 2420	Flatbed	CIS	1200

Since the proposed algorithm uses features of sensor noise, it should be able to identify the forgeries irrespective of the image-content. To examine this, the same image is scanned by using two different scanners S_4 and S_5. The forged image shown in Figure 3.13(c) is then generated by joining right half of S_4's image with left half of S_5's image . Figure 3.13(e) shows the result of applying the proposed forgery detection algorithm. The image is identified as coming from scanner S_4 with the region masked in red as the forged region. Thus the algorithm looks for differences in how the regions of an image are generated and not on the image content. One limitation with this approach is that for a similar forgery made by copying and pasting regions within the same image, the algorithm declared it as authentic image even though the forgery was visibly evident.

Figures 3.13(d), 3.14(c) and 3.14(d) show four other forgeries made by manipulating the contents of images scanned using scanner S_4. The original images corresponding to these forgeries are shown in Figures 3.13(b), 3.14(a) and 3.14(b), respectively. Figures 3.13(f), 3.14(e) and 3.14(f) show the results of applying the proposed forgery detection algorithm on these images.

Forgeries shown in Figures 3.15(c) and 3.15(d) are made by manipulating the contents of images scanned using scanner S_3. The original images corresponding to

(a) Original image

(b) Original image

(c) Forged Image

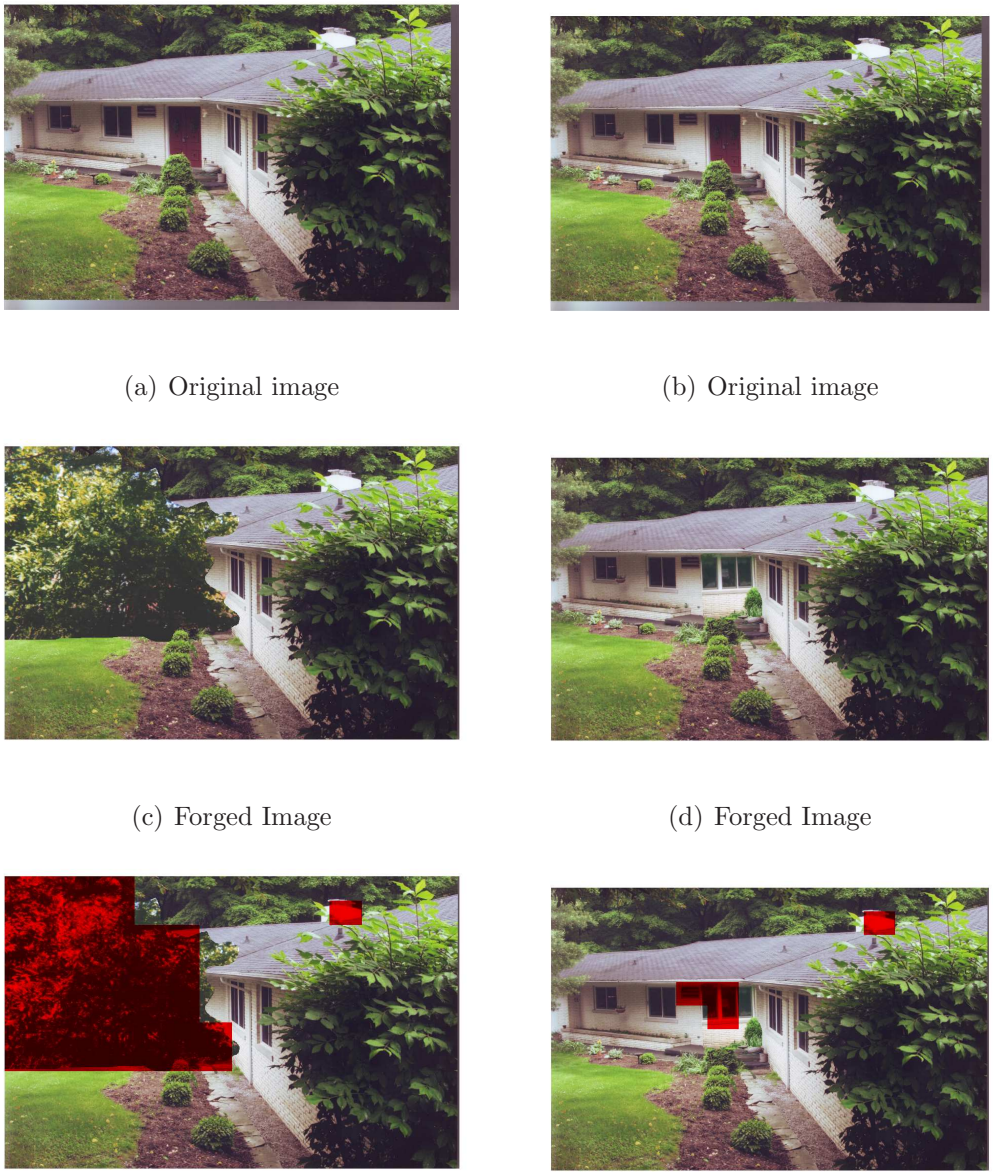
(d) Forged Image

(e) Result of Forgery Detection

(f) Result of Forgery Detection

Fig. 3.13. Results of Proposed Forgery Detection Algorithm (images in left column correspond to original image-1 and those in right column correspond to image-2).

these forgeries are shown in Figures 3.15(a) and 3.15(b), respectively. Corresponding



(a) Original image

(b) Original image

(c) Forged Image

(d) Forged Image

(e) Result of Forgery Detection

(f) Result of Forgery Detection

Fig. 3.14. Results of Proposed Forgery Detection Algorithm (images in left column correspond to original image-3 and those in right column correspond to image-4).

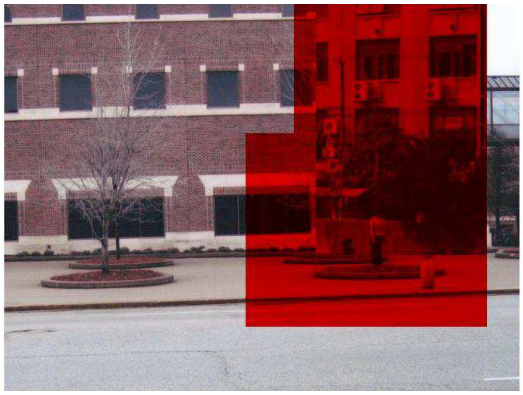
results obtained after applying the proposed scheme are shown in Figures 3.15(e) and 3.15(f).

(a) Original image

(b) Original image

(c) Forged Image

(d) Forged Image



(e) Result of Forgery Detection

(f) Result of Forgery Detection

Fig. 3.15. Results of Proposed Forgery Detection Algorithm (images in left column correspond to original image-5 and those in right column correspond to image-6).

The limitations in identifying forged regions due to the use of finite non-overlapping window are clear from these results. Further, most of the wrong classification is in the heavily textured or saturated regions.

4. SOURCE SCANNER IDENTIFICATION FROM TEXT DOCUMENTS

In the previous chapter, techniques for source scanner identification from scanned photographs were presented. These were based on using statistical features of sensor pattern noise [43] which were estimated by using a set of denoising filters. These methods for source scanner identification focused on scanned versions of images and not on scanned versions of printed text documents. Scanned documents generally lack presence of continuous tones and are dominated by "saturated" pixels. Two principle reasons which prevent direct application of this method (Chapter 3) to scanned documents are:

- The methods utilizing sensor pattern noise for source identification mainly use Photo-Response Non-uniformity (PRNU) as the sensor's signature and the PRNU is almost absent in "saturated" (completely dark or white) regions of an image [17], while the printed documents are expected to mainly have black or white pixels.

- For documents scanned at low resolution such as 200 DPI (which is generally the case for normal office usage), each character is very small, about 15×20 pixels and is non-convex, so it is difficult to filter the image in either the pixel or transform domain if we are interested only in the printed region of each character.

This chapter presents methods for authenticating scanned text documents, that have been captured by flatbed desktop scanners. Given a digital image of a text document scanned with an unknown source, henceforth referred to as the *unknown scanned document*, the goal is to identify the scanner used for generating a scanned

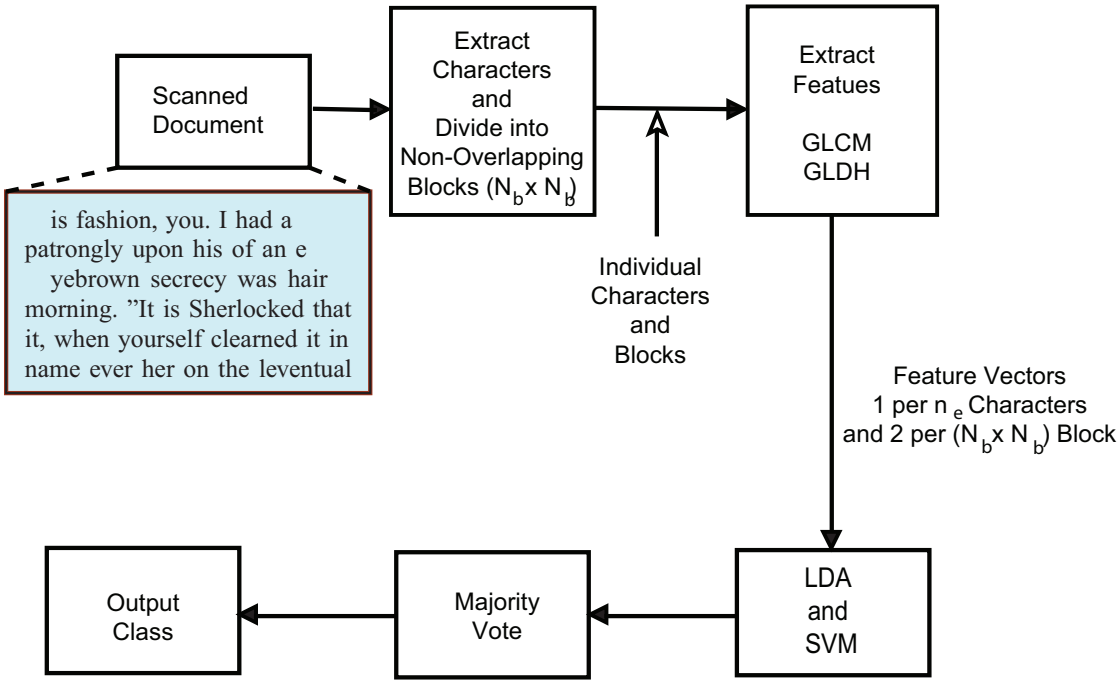


Fig. 4.1. System Diagram of Scanner Identification System.

(digital) version of the printed (hard-copy) document. Texture analysis based features are used to identify the source scanner.

4.1 System Overview

The block diagram of the proposed scanner identification system is shown in Figure 4.1. Two sets of features: character-level and block-level, are extracted for each scanned document. The first step is to extract all the letters "e" in the document. Letter "e" is the most frequently occurring character in the English language. A set of features are extracted from each group of n_e characters ("e"s) forming a feature vector for each group of n_e "e"s in the document. Further, block level features are obtained by dividing the unknown scanned document into non-overlapping blocks of size $N_b \times N_b$. A different set of features are extracted from each of these blocks. Each of these feature vectors are then classified independently using different classifiers

for each feature set. The classifiers used are a combination of Linear Discriminant Analysis (LDA) for dimensionality reduction and Support Vector Machine (SVM) for final class labeling. Let Ψ be the set of all scanners $\{S_1, S_2, \cdots, S_n\}$ (in this study Ψ is the set of 5 scanners shown in Table 4.1). For any $\phi \quad \epsilon \quad \Psi$, let $c(\phi)$ be the number of feature vectors obtained from a particular scanned document and classified as generated by scanner ϕ. The final classification is done by choosing ϕ such that $c(\phi)$ is maximum. In other words, a majority vote is performed on the resulting classifications from the SVM classifier.

4.2 Graylevel Co-Occurrence Matrix (GLCM) Features

In contrast to scanned images, scanned documents generally lack presence of continuous tones and are dominated by "saturated" pixels. In other words, most of the pixel values are either close to zero or to 255. This makes it very difficult to accurately use the type of signatures earlier used for source camera forensics [17] or for scanner identification from images [43]. For example, pattern noise (such as Photo-Response Non-uniformity, PRNU) can not be used due to it's absence in "saturated" image regions [17]. Thus, a different set of features is needed to describe each scanner uniquely.

The proposed features are based on the observation that depending upon the quality of the scanner, (i.e., it's sensitivity to sudden changes in gray-levels), the quality of edges in scanned documents will vary. More specifically, for a higher quality scanner, characters will be represented by "more" solid black lines and the transition from black to white will be sharper; and on the other hand, for a lower quality scanner, the black lines representing the characters will have more variations within them from black to lower gray levels and the transitions from black to white pixels will also be more gradual. This will result in changes in the texture features. These differences are quantified by extracting features from individual scanned characters, in particular "e"s. The graylevel fluctuation in the scanned characters in the process

direction can be modeled as textures [73]. The proposed scheme uses graylevel co-occurrence texture features as described in [73] as well as two pixel based features. This class of features are very robust for identifying printed documents [73]. Further, to alleviate problems due to the very small size of individual characters and gather sufficient statistics to estimate the Gray-Level Co-occurrence Matrix (GLCM), these matrices are generated from a group of n_e "e"s at a time. In our experiments, n_e is chosen to be 100.

Graylevel co-occurrence texture features assume that the texture information in an image is contained in the overall spatial relationships among the pixels in the image [73]. This is done by first determining the Graylevel Co-occurrence Matrix (GLCM), which is an estimate of the second order probability density function of the pixels in the image. The features are then the statistics obtained from the GLCM.

We assume that the texture in a document is predominantly in the process direction (that is, scan direction) as the same linear sensor is translated horizontally by a mechanical system to generate the complete scan. Figure 4.2 shows an idealized character, $Img(i, j)$, from which features are extracted. The region of interest (ROI) is the set of all pixels within the rectangular bounding box around the character. The determination of these bounding boxes is done by using the open source OCR system Ocrad [74].

The Gray-Level Co-occurrence Matrix (GLCM), defined in Equation 4.1, has entries $glcm(n, m, dr, dc)$ which are equal to the number of occurrences of pixels with graylevels n and m respectively with a separation of (dr, dc) pixels (Figure 4.2). If the GLCM is normalized such that its entries sum to one, the entries then represent the probability of occurrence of pixel pairs with graylevels n and m with separation (dr,dc). For generating features from each character (character level features), dc and dr are chosen to be 0 and 1 respectively.

$$\begin{aligned} &glcm(n, m, dr, dc) = \\ &\textstyle\sum_{(i,j),(i+dr,j+dc)\epsilon ROI} 1_{\{Img(i,j)=n, Img(i+dr,j+dc)=m\}} \end{aligned} \tag{4.1}$$

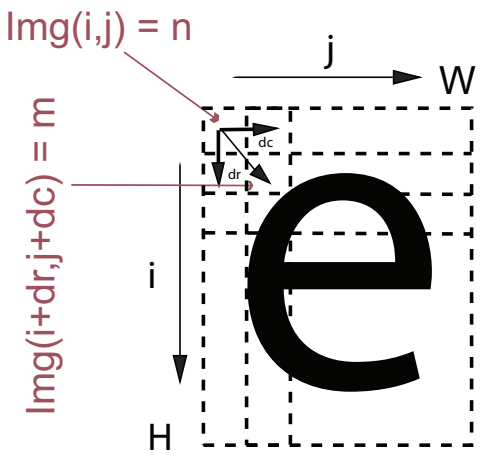


Fig. 4.2. Idealized Character for generation of $glcm(n, m)$.

The details of extracting GLCM based features is described in Appendix B. Using the equations described there, we obtain a twenty two dimensional feature vector for each input image block for specified dr and dc.

These GLCM metrics are estimated for each of the extracted "e"s and average GLCM is obtained for each group of n_e "e"s. The twenty two statistical features extracted from each of these average GLCMs are same as those used for printer identification [73]. For printer identification, hard-copy document is available to the forensic examiner. Thus, for printer identification the test document is scanned at very high resolution (such as 4800 DPI) [73]. In contrast to printer identification application [73], due to very small of size characters for 200 DPI scans, using features from GLCM's corresponding to each character separately does not provide good classification results as demonstrated by our initial experiments.

These twenty two features from the anisotropic GLCM (corresponding to dr =1, and dc =0) are extracted from each group of n_e "e"s and separately from each non-overlapping block of $N_b \times N_b$ pixels.

4.3 Modeling Edge Color Transitions

To quantify the edge transitions from 0 (black) to 255 (white), isotropic gray-level difference histogram (GLDH) are used. For each non-overlapping block of $N_b \times N_b$ pixels, in addition to 22-dimensional GLCM features, 246-dimensional isotropic gray-level difference histogram (GLDH) is also used as scanner signature. The isotropic GLDH with $d = 1$ is defined in Equations 4.2 and 4.3 (where $glcm(n, m, dr, dc)$ is in Equation 4.1). Note that in defining the isotropic GLDH, lower values of k are not used and so range of k is $[10, 255]$. These lower values of k will correspond to completely black or completely white regions and so are not useful as scanner signature and will also vary from block to block depending upon what percentage of block's area corresponds to the background. The isotropic GLDH defined in Equation 4.3 is normalized to have sum equal to one before using it as scanner signature.

$$glcm_{isotropic}(n, m) = \sum_{dr=-1}^{1} \sum_{\substack{dc=-1 \\ (dr,dc)\neq(0,0)}}^{1} glcm(n, m, dr, dc) \tag{4.2}$$

$$gldh_{isotropic}(k) = \sum_{\substack{0\leq n\leq 255 \\ 0\leq m\leq 255 \\ |n-m|=k}} glcm_{isotropic}(n, m), \;\; k \in [0, 255] \tag{4.3}$$

Hence, corresponding to an unknown scanned document with N_e "e"s and of size $N \times M$ pixels, $\left\lfloor \frac{N_e}{n_e} \right\rfloor$ 22-dimensional GLCM based features are obtained for each group of "e"s. Furthermore, $\left\lfloor \frac{(N\times M)}{(N_b\times N_b)} \right\rfloor$ 22-dimensional GLCM-based features and $\left\lfloor \frac{(N\times M)}{(N_b\times N_b)} \right\rfloor$ 246-dimensional GLDH based features are obtained for each block of size $N_b \times N_b$ pixels. The final decision about source scanner is taken by majority voting over $\left\lfloor \frac{(N\times M)}{(N_b\times N_b)} \right\rfloor + 2 \left\lfloor \frac{(N\times M)}{(N_b\times N_b)} \right\rfloor$ individual decisions.

4.4 Experimental Results

For generating testing and training datasets, the Forensic Monkey Text Generator (FMTG) (described in [75]) is used to create random documents with known statistics. Using the FMTG, it is estimated that in a page of English text printed at 10-point font

there are on average 630 “e”s [75]. For our experiments, 25 test documents (generated using FMTG, at 10-point Times New Roman font) are printed with a consumer quality laser printer (HP Laserjet 3800dn). All the documents are printed on similar quality paper and using the same printer to make sure that we are addressing the variability due to the scanners rather than the variation in paper quality or printer. The 25 test documents are scanned at 200 DPI using each of the five scanners shown in Table 4.1. To meet the requirements of most common usage, the pages are scanned at low resolution (200 DPI) with 8 bits/pixel (grayscale). In the experiments, n_e is chosen to be 100 and N_b is chosen to be 512. Thus for each of these documents of A4 size, scanned at 200 DPI, there are approximately 6 character-level feature vectors and approximately 2×12 block-level feature vectors.

Three separate classifiers (LDA + SVM) are trained for each class of features, namely GLCM features from groups of “e”s, GLCM features from each of the blocks of size $N_b \times N_b$ and isotropic GLDH features from each of the blocks of size $N_b \times N_b$. The character-level classifier (using 22-dimensional feature vector from each group of n_e “e”s) is trained with randomly chosen 375 known feature vectors and tested over a different set of 375 feature vectors. The training and testing sets are made up of 75 feature vectors from each of 5 scanners listed in Table 4.1. Two block level classifiers (one using 22-dimensional GLCM feature and another using 246 dimensional isotropic GLDH) are trained with randomly chosen 750 known feature vectors and tested over a different set of 750 feature vectors. The training and testing sets are made up of 150 feature vectors from each of 5 scanners listed in Table 4.1. The classifiers for each of these feature vectors are independent of one another. The classifier training and testing phases are repeated 100 times to obtain the final performance measures.

Figure 4.3 shows portions of the sample images scanned with different scanners. It can be seen that in some cases these images are visually differentiable due to changes in brightness and contrast settings. An unknown document might not be scanned at default scanner settings and the used brightness and contrast settings might be unknown. Therefore, before source scanner identification, the images are

Table 4.1
Scanner Set Used for Evaluation of Method for Scanner Identification using Scanned Documents.

	Make/Model	Sensor	Native Resolution (DPI)
S_1	Epson 4490 Photo	CCD	4800
S_2	OpticSlim 2420	CIS	1200
S_3	Canon LiDE 25	CIS	1200
S_4	Canon LiDE 70	CIS	1200
S_5	Canon LiDE 100	CIS	2400

pre-processed to be visually more similar by adjusting the parameters of a linear intensity transform. This will help to ensure that the proposed system will work even when the documents are scanned with different brightness and contrast settings or latter post-processed by linear intensity transformations.

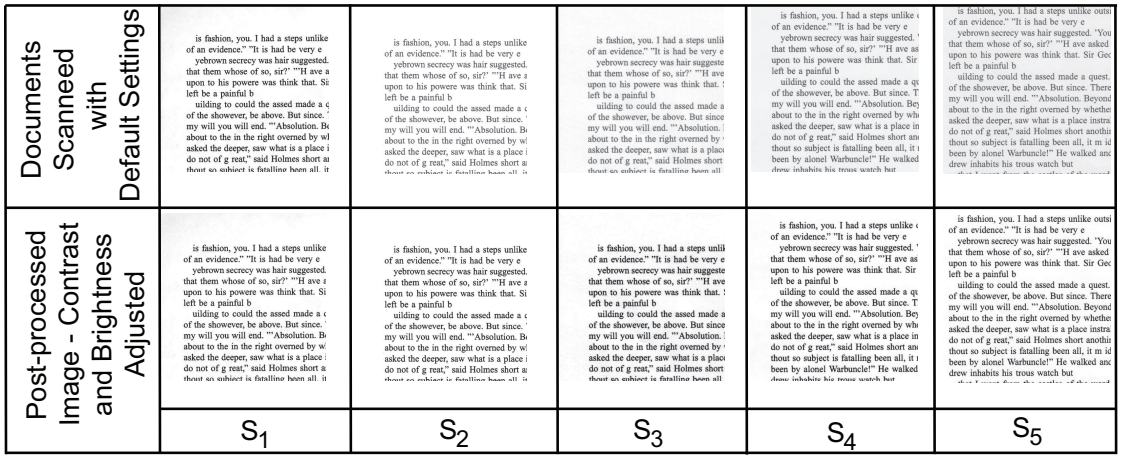


Fig. 4.3. Portions of Sample Documents from Different Scanners.

To demonstrate the efficacy of proposed features in source scanner identification, we plotted two-dimensional scatter plots showing the separability of these five scanner

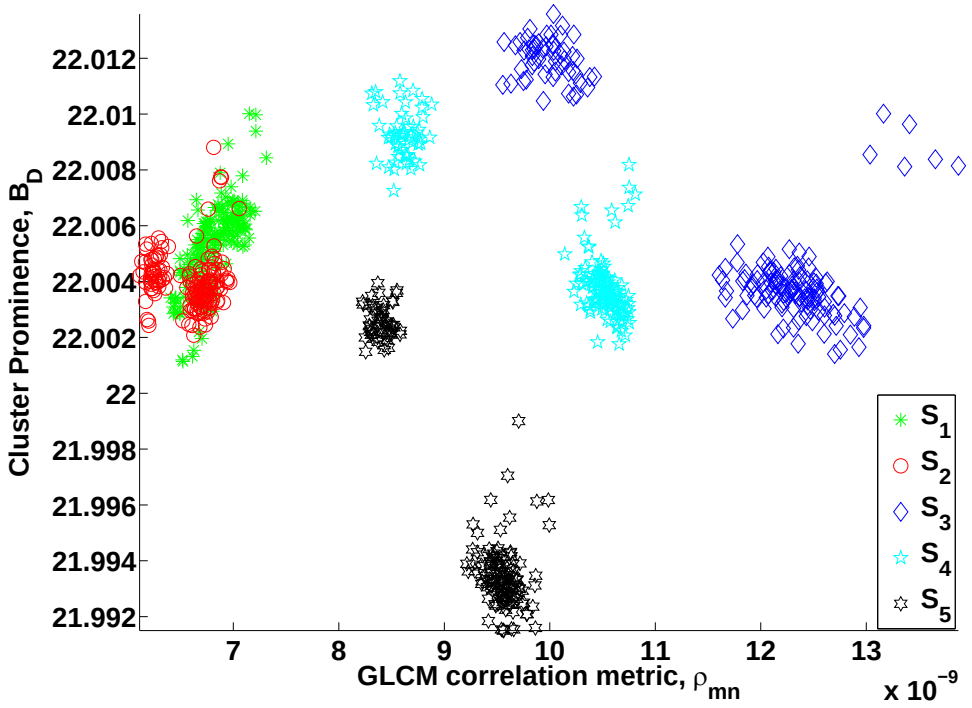


Fig. 4.4. Scatter Plot for Two Manually Chosen Character Level Features (giving best separation in 2-D feature space) of TIFF Images.

classes in low-dimensional feature space. Figure 4.4 shows the scatter plot for two manually chosen character-level features of scanned images saved in TIF format. Even though all the classes do not separate completely, the two features still have good discrimination capability. The efficacy is more evident after using Linear Discriminant Analysis (LDA) on 22-dimensional character level features and projecting them into a 7-dimensional feature space. Figure 4.5 shows scatter plots for the two projected features with maximum discrimination.

Table 4.2 shows the average accuracy of the dedicated classifiers for scanned documents saved in different formats. The classifiers are trained and tested on feature vectors coming from scanned documents saved in the same format. Note that the accuracy values pertain to classification of individual feature vectors and not the complete document. In all these cases, the accuracy for classifying complete docu-

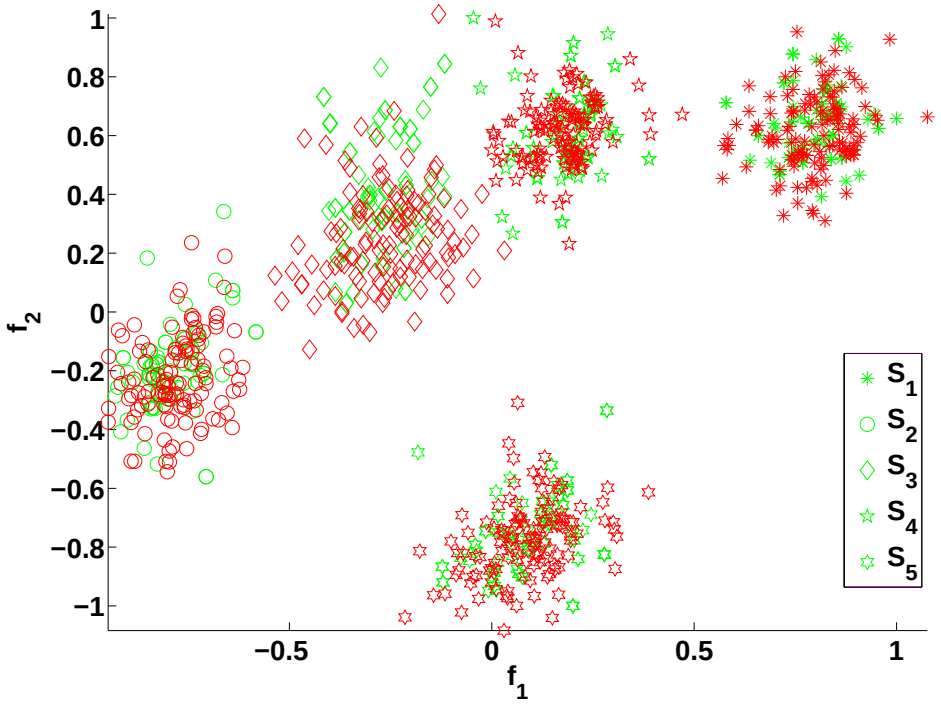


Fig. 4.5. Scatter Plot for Two Manually Chosen Character Level Features (after performing LDA) of TIFF Images, (green symbols correspond to the feature vectors used for training LDA and red corresponds the feature vectors used for testing).

ments is 100%. To see the effectiveness of the proposed scheme in scenarios where the JPEG quality factor may not be reliably known or estimated, another set of three general classifiers are trained and tested on randomly chosen feature vectors from images saved with two different JPEG quality factors (Q =80 and 60). Both training and testing sets include features from documents saved at different quality factors. Table 4.3 shows the confusion matrix for block-level isotropic GLDH features which has average classification accuracy of 98%. Similar general classifiers for the character-level GLCM statistics and block-level GLCM statistics have average accuracy values of 99.7% and 99.4% respectively. In all our experiments, after majority

voting the source scanner amongst five scanners is found with 100% classification accuracy.

Table 4.2

Average Accuracies of Dedicated Classifiers for Scanner Identification Using Scanned Documents.

Image Format	Feature Type	Average Accuracy
TIFF	Character Level GLCM	99.9
	Block Level GLCM	99.9
	Block Level GLDH	96.4
JPEG (Q =80)	Character Level GLCM	99.7
	Block Level GLCM	99.7
	Block Level GLDH	98
JPEG (Q =60)	Character Level GLCM	99.6
	Block Level GLCM	99.5
	Block Level GLDH	95.2

Table 4.3

Confusion Matrix for General Classifier (testing and training on JPEG images with Q =80 and 60).

		Predicted				
		S_1	S_2	S_3	S_4	S_5
Actual	S_1	96.9	2.9	0	0.0	0.2
	S_2	3.4	96.6	0	0	0.0
	S_3	0	0	98.2	1.0	0.8
	S_4	0	0	0.4	99.5	0.1
	S_5	0	0	0.3	0.2	99.5

5. IMAGE SOURCE CLASSIFICATION

This chapter presents methods for image source classification for forensic applications. That is, given a digital image of unknown origin, the aim is to assign it to one of the three classes:

- Digital Camera Generated (CG) images,
- Scanner Generated (SG) images, and
- Photorealistic Computer Generated (PRCG) images

In this dissertation, the term PRCG is used for computer generated images which appear to be real images (photographs). This implies that it excludes other computer graphics such as icons, buttons, graphs, which are easily distinguished from photographs. These image classes are decided in terms of "last" system in the processing/creation chain of an image and not on the basis of image content. So, a photograph of a printed version of computer generated image falls within the class CG. Similarly, the scanned versions of either printed real scenes or printed PRCG images belong to class SG. In the algorithms that we developed, it is assumed that the images are from single source and are not a mosaic of sub-images from different sources. Research work on related problems in other fields of image classification include differentiating city images from landscape images [76], indoor images from outdoor images [77], photographs from paintings [78], photographs from (non-realistic) graphical icons [79] and techniques for evaluating the photo-realism of computer graphics rendered images from a human perception point of view [80].

There are a number of methods proposed for solving the image source classification problem (Section 2.2). Although all these methods differ in details, extraction of suitable features and use of a classifier for recognizing common pattern amongst

these features is the common fabric behind all these methods. The features used for classification vary from method to method and in one sense the search for optimal feature set is the aim of this branch of image forensics. A classifier trained from the features of images with known origin is used to classify an image of unknown origin. Most of these methods use SVM for classification, some of them also use LDA to improve the performance and visualization of features. These features are derived from differences in image generation techniques used by the three systems and from the gross or subtle differences in the image content of a real and computer generated image. Hence, accuracy and reliability of various methods depends upon characterization of source class dependent features, features common amongst all scanners or all cameras or all computer rendering softwares. These features are expected to be "orthogonal" to the features which are successfully used for source camera identification [2, 15, 17] or source scanner identification [26, 37, 43]. This is because, in case of source camera identification or source scanner identification, we are interested in features which differ from camera to camera or from scanner to scanner and for the present problem we need to identify the features common to all the cameras or all the scanners. The common limitation of this class of methods is that given the knowledge of the features used by a particular method, it is almost always possible to come up with suitable post-processing steps which will prevent the successful source classification. For example, to prevent correct detection by color filter array and demosaicing based methods, one can re-sample and re-interpolate a given image using another demosaicing algorithm.

5.1 Feature Vector Selection

Both digital cameras and scanners work on a similar principle in terms of the imaging pipeline. However, digital cameras use a two dimensional sensor array while most scanners use a one dimensional linear array. In the case of flatbed scanners, the same linear array is translated to generate the entire image. It is expected to

find periodic correlation between rows of the fixed component of the sensor noise (Section 1.2.4) of a scanned image. There is no reason to find a similar periodic correlation between columns of the sensor noise of a scanned image. Neither the rows nor the columns of the fixed component of the sensor noise of an image generated by a digital camera are expected to exhibit such periodicity. This difference can be used as a basis for discriminating between the two source classes, SG and CG. Further, due to the fundamental differences in the image generation process, the residual noise in computer generated images may not have properties similar to those of images from the other two classes.

Inspired by the success of statistical features of pattern noise for source scanner identification (Chapter 3), the 204-dimensional features mentioned in Section 3.2.1 are used here for image source classification.To reduce the dimensionality of the feature vectors, LDA [66] is used and a five dimensional feature vector is obtained for each image. Each component of the five dimensional feature vector is then a linear combination of the original 204 features. Finally a SVM classifier is used to classify these five dimensional feature vectors.

5.2 Experimental Design

Table 5.1 shows the sources of different classes of digital images used in our experiments. Computer generated images include images from number of different methods such as 3ds max, Maya, Softimage and Lightwave. Computer generated images, in JPEG format, were downloaded from publicly available websites listed in Table 5.1. For computer generated images of varying sizes, a central 1024×768 or 512×512 block is used for feature extraction depending upon the size of the image. 350 images were captured from each of the three cameras at 1024×768 resolution and stored in the best quality JPEG format supported by each camera. Some of the scanners have CCD sensor while others have CIS sensor. Scanned images are generated at two different scanning scenarios. Under first scenario, approximately 30 images are

Table 5.1
Image Sources Used for Evaluation of Image Source Classification Method.

Image Class	Devices Used
Digital Camera CG	Canon PowerShot SD200, Nikon Coolpix 4100, Nikon Coolpix 7600
Computer Generated PRCG	www.3dlinks.com, www.irtc.org, www.raph.com, www.digitalrepose.com, www.maxon.net, www.realsoft.com
Flatbed Scanners SG	Epson Perfection 4490 Photo, HP ScanJet 6300c-1, HP ScanJet 6300c-2, HP ScanJet 8250, Mustek 1200 III EP, Visioneer OneTouch 7300, Canon LiDe 25, Canon Lide 70, OpticSlim 2420, Visioneer OneTouch 7100, Mustek ScanExpress A3

scanned from each of the 11 scanners (2 out of 11 are of the same model) at the native resolution of the scanners. That gives us images at 1200 DPI or 4800 DPI. The images are then sliced into blocks (sub-images) of size 1024×768 pixels and sub-images from the first two columns of the scanned images are used. Under second scenario, from each of the 11 scanners 108 images were scanned at 200 DPI resolution and stored in TIF format (1024×768 pixels). Hence in total, we have 1000 PRCG images, 1050 CG images, 1800 SG sub-images (from images scanned at native resolution) and 1000 SG images (scanned at 200 DPI). Figure 5.1 shows a sample of the images used in this study.

The `LIBSVM` package [67, 68] is used in this study for the SVM classifier. A radial basis function is chosen as the kernel function and grid search is performed to select the best parameters for the kernel. Unless stated otherwise, randomly chosen 80% of the images are used for training the classifier and rest of the images are used for testing. This training and testing is repeated multiple times to obtain the final average classification results.

(a) CG

(b) CG

(c) SG

(d) SG

(e) PRCG

(f) PRCG

Fig. 5.1. Sample Images Used in Experiments on Image Source Classification.

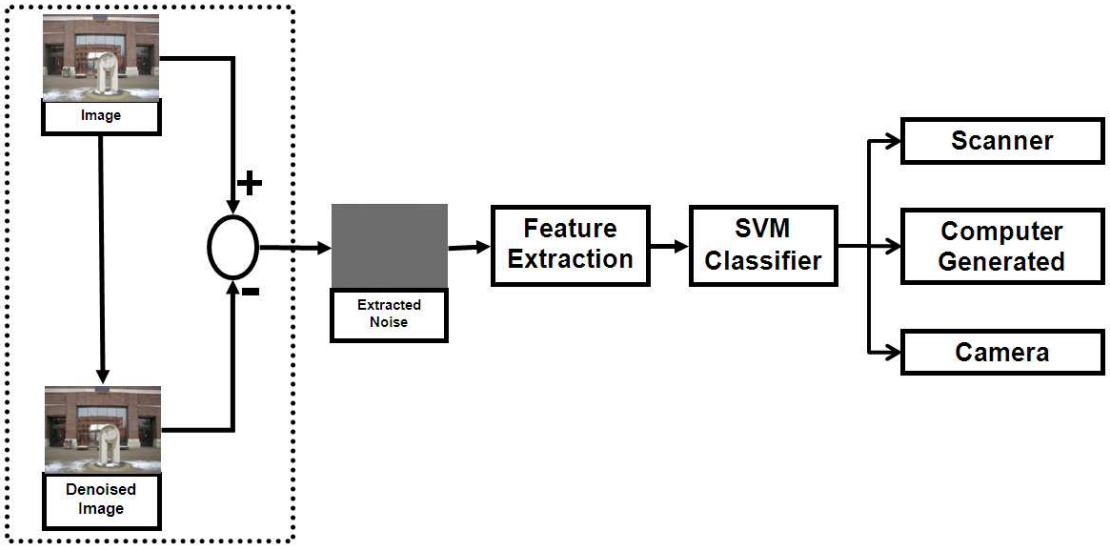


Fig. 5.2. Block Diagram of Image Source Classification Method.

The complete experimental protocol is shown in Figure 5.2. To check the effectiveness of our proposed scheme in classifying images based on their sources, a number of experiments are performed by varying the type of images and the number of features used for classification.

5.3 Experimental Results: Native Resolution Scanned Images

In our initial experiments on differentiating images scanned at native resolution of the scanners from the digital camera images, we used selected statistical features obtained from noise estimated by "LPA-ICI" based denoising filter, instead of complete 204-dimensional features from the denoising filterbank.

5.3.1 Training Using the Complete Dataset

In this experiment 1800 sub-images from three scanners and 1050 images from three digital cameras are used. Out of the 2850 images, half are randomly chosen

to train the SVM and rest are used for testing. Initially, the feature vectors are generated using only mean, median, mode and ratio of average values of ρ_{row} and ρ_{col} (7-dimensional feature vectors for each image).

An average classification accuracy of 98.1% is obtained over multiple runs in this case, and the confusion matrix is shown in Table 5.2. To improve the classification accuracy, another SVM model is generated using feature vectors containing the first as well higher order statistics of ρ_{row} and ρ_{col}. In this case, an average classification accuracy of 98.6% is obtained. The corresponding confusion matrix is shown in Table 5.3.

Table 5.2
Native Resolution TIFF Sub-images Using 7 Dimensional Feature Vector

		Predicted	
		Scanner	Camera
Actual	Scanner	97.9	2.1
	Camera	1.6	98.4

Table 5.3
Native Resolution TIFF Sub-images Using 17 Dimensional Feature Vector

		Predicted	
		Scanner	Camera
Actual	Scanner	98.4	1.6
	Camera	1.2	98.8

5.3.2 Training Without "Saturated" Images

In completely white or completely black images (henceforth referred to as "saturated" images) the sensor noise is highly suppressed [28, 36]. Because the proposed method of imaging sensor classification utilizes features from the fixed component of the sensor noise, such saturated images are likely to be mis-classified. The images mis-classified in Section 5.3.1 show that this is indeed the case. In this experiment, the saturated images are removed from the dataset, which leaves a total of 2000 scanned and non-scanned images. Many sub-images from the scanned images come under this excluded category because they are portions of bright areas (sky) and dark areas (roads) of the full images.

As before, half the images are chosen randomly for training and the other half for testing. Using only the first order statistics of ρ_{row} and ρ_{col}, an average classification accuracy of 98.9% is obtained with the confusion matrix shown in Table 5.4. Using the first as well higher order statistics of ρ_{row} and ρ_{col}, an average classification accuracy of 99.3% is obtained. The corresponding confusion matrix is shown in Table 5.5.

Table 5.4

Native Resolution TIFF Sub-images Using 7 Dimensional Feature Vector (excluding the "saturated" images)

		Predicted	
		Scanner	Camera
Actual	Scanner	98.7	1.3
	Camera	0.9	99.1

Table 5.5

Native Resolution TIFF Sub-images Using 17 Dimensional Feature Vector (excluding the "saturated" images)

		Predicted	
		Scanner	Camera
Actual	Scanner	99.2	0.8
	Camera	0.6	99.5

5.3.3 Restricted Training

To evaluate the robustness of the proposed scheme in a situation when the imaging device to be tested is unavailable for training, SVM classifier is trained using features from images captured by the HP ScanJet 6300c-1, HP ScanJet 6300c-2, Canon Powershot SD200 and Nikon Coolpix 4100, while the testing set includes images from the Epson Perfection 4490 and the Nikon Coolpix 7600 as well. Using first as well as higher order statistics of ρ_{row} and ρ_{col}, an average classification accuracy of 93.5% is obtained with the corresponding confusion matrix shown in Table 5.6. In a similar experiment in which the HP Scanjet 6300c-1 and Canon Powershot SD200 are not used for training, an average classification accuracy of 93.67% is obtained with the corresponding confusion matrix shown in Table 5.7.

5.3.4 Effect of JPEG Compression

The efficacy of the proposed scheme is also tested on images that have been JPEG compressed. An average classification accuracy of 93.5% is obtained for JPEG images compressed using quality factor 90, as shown by the confusion matrix in Table 5.8. Both the training and testing images are JPEG compressed at quality factor 90.

Table 5.6

Native Resolution TIFF Sub-images Using 17 Dimensional Feature Vector, Trained Without Images from Epson 4490 and Nikon Coolpix 7600

		Predicted	
		Scanner	Camera
Actual	Scanner	98.1	1.9
	Camera	10.9	89.1

Table 5.7

Native Resolution TIFF Sub-images Using 17 Dimensional Feature Vector, Trained Without Images from HP Scanjet 6300c-1 and Canon Powershot SD200

		Predicted	
		Scanner	Camera
Actual	Scanner	98.5	1.5
	Camera	11.2	88.8

Table 5.8
Native Resolution JPEG (Q=90) Sub-images Using 17 Dimensional Feature Vector

		Predicted	
		Scanner	Camera
Actual	Scanner	97.6	2.4
	Camera	7.1	92.9

5.4 Experimental Results: Non-native Resolution Scanned Images

After seeing the efficacy of proposed method in classifying sub-images from images scanned at native resolution of the scanners, detailed experiments are conducted using

complete 204 dimensional features from the denoising filterbank to show the efficacy of proposed method for solving three class classification problem. These experiments differentiate images scanned at 200 DPI from digital camera images and computer generated images.

In the first set of experiments three separate SVM classifiers are designed for distinguishing between three possible pairs of image source classes: SG, CG and PRCG. Tables 5.9, 5.10 and 5.11 show the confusion matrices for classifying these pair of classes. The average classification accuracy for distinguishing scanner images from camera images is 97.6%. The average classification accuracy for distinguishing PRCG images from camera images is 91.5%. While the average classification accuracy for distinguishing camera images from scanner images is 89.4%, the lowest among three pairs.

Table 5.9
Using Statistical Features: Scanner vs. Camera (scanned images at 200 DPI)

		Predicted	
		Scanner	Camera
Actual	Scanner	98.2	1.8
	Camera	3.1	96.9

Table 5.10
Using Statistical Features: PRCG vs. Camera

		Predicted	
		PRCG	Camera
Actual	PRCG	88.3	11.6
	Camera	5.2	94.8

Table 5.11
Using Statistical Features: Camera vs. Scanner (scanned images at 200 DPI)

		Predicted	
		Camera	Scanner
Actual	Camera	89.5	10.5
	Scanner	10.7	89.3

The average classification accuracy for classifying TIFF images from all three classes is 94.7%. Corresponding confusion matrix is shown in Table 5.12. Thus, by training an SVM classifier on the 204 dimensional feature vectors from each image, using randomly chosen 800 images from each class for training and separate 200 images for testing, proposed method achieves average classification accuracy of 94.7%.

Table 5.12

Using Statistical Features: Scanner vs. PRCG vs. Camera (scanned images at 200 DPI), TIFF

		Predicted		
		Camera	PRCG	Scanner
Actual	Camera	94.3	5.1	0.7
	PRCG	6.4	91.9	1.7
	Scanner	0.6	1.0	98.4

5.4.1 Restricted Training

To check the robustness of the proposed scheme when the imaging device to be tested is unavailable for training, a SVM classifier is trained without images from scanners Epson Perfection 4490 and Visioneer OneTouch 7100 and tested on scanned

images from these scanners only. This classifier has an average classification accuracy of 94% with the corresponding confusion matrix shown in Table 5.13. All the training and testing images used by this classifier are saved in JPEG format with Q = 90. This result shows that the proposed method will classify an image scanned at 200 DPI as scanned image even if the images from that particular scanner are not available for training.

Table 5.13

Using Statistical Features: Scanner vs. PRCG vs. Camera, JPEG (Q = 90), (training without S_1 and S_{10}, scanned images at 200 DPI).

		Predicted		
		Camera	PRCG	Scanner
Actual	Scanner	1.9	4.2	94

5.4.2 Effect of JPEG Compression

The efficacy of the proposed method is also tested on images that have been JPEG compressed. First two dedicated classifiers are trained and tested for images saved in JPEG format, with quality factors Q = 90 and Q = 70. An average classification accuracy of 94.6% is obtained when all the scanned images are saved as JPEG (Q=90) before feature extraction for classifier training and testing. Corresponding confusion matrix is shown in Table 5.14. The average classification accuracy for dedicated classifier for images saved at quality factor 70, is 92.5%.

Further, a general classifier is designed by training and testing on a combined set of images saved at JPEG quality factors Q=90 and Q =70. This general classifier has a classification accuracy of . Corresponding confusion matrix is shown in Table 5.16. This slight decrease in performance is as expected since the pattern noise

Table 5.14

Using Statistical Features: Scanner vs. PRCG vs. Camera, JPEG (Q = 90) (scanned images at 200 DPI).

		Predicted		
		Camera	PRCG	Scanner
Actual	Camera	92.8	3.6	3.7
	PRCG	3.4	93.1	3.4
	Scanner	1.1	1.1	97.9

Table 5.15

Using Statistical Features: Scanner vs. PRCG vs. Camera, JPEG (Q = 70) (scanned images at 200 DPI).

		Predicted		
		Camera	PRCG	Scanner
Actual	Camera	91.9	2.7	5.4
	PRCG	6.1	87.8	6.2
	Scanner	1	1.1	97.8

degrades with JPEG compression and down-sampling, as observed for source camera identification [15] and source scanner identification [25].

Further experiments by varying the size of training dataset show that average classification accuracy remains close to 80% even when only 40% images (400 images from each source class) are used for training the classifier.

Table 5.16

Using Statistical Features: Confusion Matrix for Classifying JPEG Compressed Images (scanned images at 200 DPI).

		Predicted		
		Scanner	PRCG	Camera
Actual	Scanner	86.4	1.7	11.9
	PRCG	11.8	70.6	17.6
	Camera	13.4	4.2	82.4

6. SUMMARY AND FUTURE WORK

This thesis focused on forensic fingerprinting methods for image source classification and scanner identification (Figure 6.1). These methods can be used for any scanning scenario as long as training images from the concerned devices are available. This chapter briefly summarizes the salient contributions of this dissertation and proposes some directions for future developments.

6.1 Summary

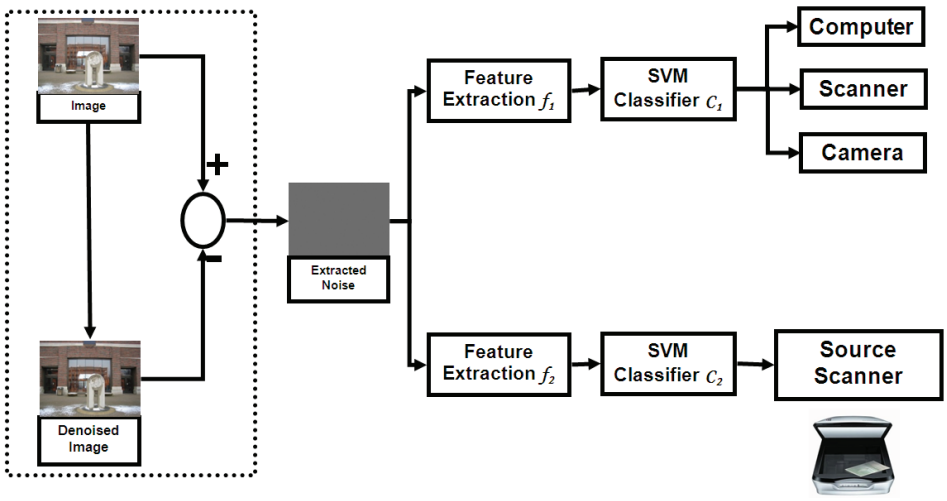


Fig. 6.1. Image Forensics using Statistical Features of Sensor Noise.

Selection of proper features is the key to achieve accurate results for image forensic applications. None of the existing forensic methods, including those presented here, are independently capable of solving the image forensic problem in its entirety and under all circumstances and it is possible to design suitable attacks to prevent them

from correct classification. But, when combined together as a set of tools, they can achieve a high degree of accuracy in a wide range of scenarios. The main contributions of this dissertation are as follows:

- **Verification of Sensor Noise-based Camera Identification Scheme**: As a first step towards development of new methods for different problems in scanner forensics, we performed extensive experiments for verification of sensor noise based camera forensic method [2, 15, 17]. The results of these independently conducted experiments on a completely different set of cameras, were similar to those reported in earlier papers.

- **Source Scanner Identification from Scanned Images**: We investigated the use of imaging sensor pattern noise for source scanner identification and compared the end-to-end system performance with other existing methods. These results presented in Section 3.4 show that the statistical feature vector based method gives high accuracy for source scanner identification, both for native resolution and lower resolution scanned images. The results in Table 3.11 indicate that it is possible to discriminate between scanners of the same make and model for images scanned at native scanning resolution. On the other hand, for images scanned at lower non-native resolutions such as 200 DPI, the proposed scheme successfully identifies the scanner make and model, and groups scanners of the same make and model into a single class (Table 3.19). For scanner model identification among eleven scanners of ten different models, an average classification accuracy of 99.9% is obtained. Table 3.24 and 3.27 show that the proposed scheme performs well even with images that have undergone JPEG compression with low-quality factors, image sharpening, and contrast stretching.

- **Forgery Detection in Scanned Images**: We extended the use of statistical features of image sensor pattern noise for forgery detection in scanned images. Results shown in Figures 3.13, 3.14, and 3.15 show the efficacy of this method

for identifying forgeries in images scanned at native resolution of the scanners. The limitation on minimum size of forged regions that can be identified with this approach depends upon the size of sliding window. To maintain the statistical significance of the features used for classification, we can not use window sizes below a certain threshold. This threshold is experimentally determined to be 384×512 pixels for images scanned at 1200DPI, i.e. approximately 0.4×0.5 inches. Further, results show that the proposed algorithm identifies forgeries independent of image content, but fails to identify them when made by copying and pasting regions within the same image. Thus, the proposed scheme can be an effective tool for forgery detection in scanned images, if used in co-ordination with other existing methods for forgery detection.

- **Source Scanner Identification from Documents**: We proposed methods for source scanner identification for scanned text documents using texture features (Chapter 4). As shown by the experiments (Section 4.4, Tables 4.2 and 4.3), the proposed method is robust to JPEG compression and gives 100% classification accuracy for classifying A4 size text documents and more than 95% classification accuracy for classifying smaller blocks of size 512×512 or a group of 100 "e"s.

- **Imaging Source Classification**: Use of the sensor pattern noise for classifying digital images based on their originating mechanism, a scanner or a digital camera or a computer graphics algorithm, is investigated (Chapter 5). The proposed scheme utilizes statistical properties of residual noise and difference in the geometry of imaging sensors, and demonstrates promising results. As shown by results (Sections 5.3.3 and 5.4.1), the proposed scheme does not need the actual source device for training purposes. Thus, images generated by even a completely unknown scanner or digital camera can be classified properly.

6.2 Future Work

The research problems addressed in this dissertation are currently of active interest in sensor forensics. Some specific ideas for extending the present work are as follows:

- For source scanner identification different classifiers need to be designed for different scanning resolutions. In other words, to test an image scanned at 200 DPI, we need to train a classifier with known images scanned at 200 DPI; another classifier trained on images scanned at 1200 DPI can not be used. It is an interesting problem to remove this constraint by modeling the relationship between the statistical features and scanning resolution. A universal classifier can then be designed to work with all the scanning resolutions. This will also reduce the requirements on training image dataset.

- Presently classifiers designed for source scanner identification using scanned documents are dependent on font type and size. Modeling the relationship between proposed features, and font type and size will be very useful. Extensive experimentation can be done to see the effect of variation in font type and size. Scalability of these forensic techniques need to be tested on bigger datasets.

- Forgery detection currently determines whether an image is authentic and also identifies the source scanners of different portions of a scanned image. It requires training images from different scanners to build the classifier. It would be interesting to modify this method in such a way that it does not need to train a classifier. Instead, user input can be obtained for portions of an image which are definitely authentic and then unsupervised learning techniques can be used to make decisions regarding authenticity of other image regions.

- Presently the forgery detection scheme searches the entire image for forged regions. User interactivity to select a region of interest which has questionable authenticity, will also reduce the computation time and improve the accuracy.

- Extensive experiments on post-processed images should be performed to examine the robustness of image source classification methods.

- Robustness Analysis and Spoofing Image Forensic Techniques: Although a number of image forensic methods have been proposed in last few years, a systematic framework to analyze them for their robustness against intentional as well as unintentional post-processing operations is lacking. Some of these operations, such as saving the image at a lower JPEG quality, are content-preserving and are harmless in most scenarios. While other operations such as changing regions in an image by copy pasting or changing the background are content-changing and their occurrence must be detected. We would like to develop mathematical models for commonly occurring image manipulation operations and investigate possible spoofing scenarios for different detection tools. For example, the method for source camera identification using sensor pattern noise as proposed in [15] uses a correlation detector for detecting the source camera. Rotating/cropping or shifting the image even by one pixel will lead to desynchronization and thus failure of this detection method. Removing the noise by using the same denoising filter is another alternative to avoid detection using this method. Similarly image forensic methods based on Color Filter Array (CFA) and demosaicing artifacts will fail if the image is resampled on a different Bayer lattice and is interpolated using different interpolation scheme, without making any change in the image content or it's appearance to human eye.

- Characterization of Video Cameras and Audio Devices: By designing suitable features, the proposed framework can be extended for source identification from video clips and audio files as well. In many situations, a video is first captured in analog form and later transfered and coded into a digital form. Features have to be designed to model the properties of the capturing device and not the encoder or analog-to-digital converter. Audio capture devices pose similar problem because the quality of audio files produced depends on the method

of recording signal from the audio device. For example, most of the audio recording systems allow a user to set a threshold for noise or to simultaneously add a background signal to the captured signal.

6.3 Publications

Journal papers:

1. P.-J. Chiang, **Nitin Khanna**, Aravind K. Mikkilineni, Maria V. O. Segovia, Sungjoo Suh, Jan P. Allebach, George T.-C. Chiu, and Edward J. Delp, "Printer and scanner forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 72–83, March 2009.

2. **Nitin Khanna**, Aravind K. Mikkilineni, and Edward J. Delp, "Scanner identification using feature-based processing and analysis," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 123–139, March 2009.

3. **Nitin Khanna**, Aravind K. Mikkilineni, and Edward J. Delp, "Forensic camera classification: Verification of sensor pattern noise approach," *Forensic Science Communications (FSC)*, vol. 11, no. 1, January 2009.

4. **Nitin Khanna**, Aravind K. Mikkilineni, Anthony F. Martone, Gazi N. Ali, George T.-C. Chiu, J. P. Allebach, and Edward J. Delp, "A survey of forensic characterization methods for physical devices," *Digital Investigation*, vol. 3, no. Supplement 1, pp. 17–28, July 2006.

Conference Papers:

1. **Nitin Khanna** and Edward J. Delp, "Source scanner identification for scanned documents," *First IEEE International Workshop on Information Forensics and Security*, London, UK, December 2009, submitted.

2. **Nitin Khanna** and Edward J. Delp, "Forensic techniques for image source classification: A comparative study," *IWDW09: 8th International Workshop on Digital Watermarking*, Guildford, UK, August 2009, to appear.

3. **Nitin Khanna**, Antoni Roca, George T. C. Chiu, Jan P. Allebach, and Edward J. Delp, "Improvements on image authentication and recovery using distributed source coding," vol. 7254, no. 1. SPIE, 2009, p. 725415.

4. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Survey of scanner and printer forensics at purdue university," *IWCF '08: Proceedings of the 2nd international workshop on Computational Forensics*. Berlin, Heidelberg: Springer-Verlag, August 2008, pp. 22–34.

5. **Nitin Khanna**, George T. Chiu, Jan P. Allebach, and Edward J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, March 2008, pp. 1653–1656.

6. **Nitin Khanna**, George T. Chiu, Jan P. Allebach, and Edward J. Delp, "Scanner identification with extension to forgery detection," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, no. 1, San Jose, CA, January 2008, p. 68190G.

7. **Nitin Khanna**, Aravind K. Mikkilineni, Pei J. Chiang, Maria V. Ortiz, Sungjoo Suh, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Sensor forensics: Printers, cameras and scanners, they never lie," *IEEE International Conference on Multimedia and Expo*, July 2007, pp. 20–23.

8. **Nitin Khanna**, Aravind K. Mikkilineni, Pei J. Chiang, Maria V. Ortiz, Vivek Shah, Sungjoo Suh, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Printer

and sensor forensics," *IEEE Workshop on Signal Processing Applications for Public Security and Forensics (SAFE 07)*, April 2007, pp. 1–8.

9. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Forensic classification of imaging sensor types," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1. San Jose, CA, USA: SPIE, January 2007, p. 65050U.

10. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Scanner identification using sensor pattern noise," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1, San Jose, CA, USA, January 2007, p. 65051K.

LIST OF REFERENCES

LIST OF REFERENCES

[1] P.-J. Chiang, N. Khanna, A. K. Mikkilineni, M. V. O. Segovia, S. Suh, J. P. Allebach, G. T.-C. Chiu, and E. J. Delp, "Printer and scanner forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 72–83, March 2009.

[2] J. Fridrich, "Digital image forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 26–37, March 2009.

[3] T.-T. Ng and S.-F. Chang, "Identifying and prefiltering images," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 49–58, March 2009.

[4] H. Farid, "Image forgery detection," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 16–25, March 2009.

[5] A. Swaminathan, M. Wu, and K. Liu, "Component forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 38–48, March 2009.

[6] The Fedral Reserve Board. (April 24, 2007) Check clearing for the 21st century act. Http://www.federalreserve.gov/paymentsystems/truncation/.

[7] D. Page. (January 2007) Forensics focuses on digital photography. Http://www.forensicmag.com/articles.asp?pid=120.

[8] D. McCullagh. (September 30, 1996) Child pornography prevention act 1996. Http://www.politechbot.com/docs/cppa.text.html.

[9] Adobe Systems Incorporated. (2009) Adobe photoshop cs4. Http://www.adobe.com/products/photoshop/photoshop/.

[10] N. Khanna, A. K. Mikkilineni, A. F. Martone, G. N. Ali, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "A survey of forensic characterization methods for physical devices," *Digital Investigation*, vol. 3, no. Supplement 1, pp. 17–28, July 2006.

[11] N. Khanna, A. K. Mikkilineni, P.-J. Chiang, M. V. Ortiz, V. Shah, S. Suh, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Printer and sensor forensics," *IEEE Workshop on Signal Processing Applications for Public Security and Forensics (SAFE 07)*, April 2007, pp. 1–8.

[12] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh, "Methods for identification of images acquired with digital cameras," *Enabling Technologies for Law Enforcement and Security*, vol. 4232, no. 1. SPIE Press, 2001, pp. 505–512.

[13] J. Lukas, J. Fridrich, and M. Goljan, "Determining digital image origin using sensor imperfections," *Proceedings of the SPIE International Conference on Image and Video Communications and Processing*, vol. 5685, no. 1. SPIE, 2005, pp. 249–260.

[14] J. Lukas, J. J. Fridrich, and M. Goljan, "Digital "bullet scratches" for images," *IEEE International Conference on Image Processing*, vol. III, Genova, September 2005, pp. 65–8.

[15] ——, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205– 214, June 2006.

[16] M. Chen, J. Fridrich, and M. Goljan, "Digital imaging sensor identification (further study)," vol. 6505, no. 1. SPIE, 2007, p. 0P0Q.

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

[18] M. Kharrazi, H. T. Sencar, and N. D. Memon, "Blind source camera identification," *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, October 2004, pp. 709–712.

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

[20] S. Bayram, H. Sencar, N. Memon, and I. Avcibas, "Source camera identification based on CFA interpolation," vol. 3, September 2005, pp. 69–72.

[21] N. Khanna, A. K. Mikkilineni, and E. J. Delp, "Forensic camera classification: Verification of sensor pattern noise approach," *Forensic Science Communications (FSC)*, vol. 11, no. 1, January 2009.

[22] N. Khanna, A. K. Mikkilineni, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Forensic classification of imaging sensor types," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1. San Jose, CA, USA: SPIE, January 2007, p. 65050U.

[23] N. Khanna, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, March 2008, pp. 1653–1656.

[24] C. McKay, A. Swaminathan, H. Gou, and M. Wu, "Image acquisition forensics: Forensic analysis to identify imaging source," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, USA, March 2008, pp. 1657–1660.

[25] T. Gloe, E. Franz, and A. Winkler, "Forensics for flatbed scanners," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1, January 2007, pp. 1–12.

[26] H. Gou, A. Swaminathan, and M. Wu, "Robust scanner identification based on noise features," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1, January 2007, pp. 1–11.

[27] A. F. Martone, A. K. Mikkilineni, and E. J. Delp, "Forensics of things," *Proceedings of the 2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, Denver, Colorado, March 2006, pp. 149–152.

[28] G. C. Holst, *CCD Arrays, Cameras, and Displays, Second Edition.* JCD Publishing & SPIE Press, USA, 1998.

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

[30] J. Tyson. (2001) How scanners work. Http://computer.howstuffworks.com/scanner.htm.

[31] PCTechGuide. (2001, November) Scanners. Http://www.pctechguide.com/55Scanners.htm.

[32] M. Mukhtar, J. Yi, and G. T.-C. Chiu, "Color registration error reduction in document scanner using iterative velocity command synthesis," *Proceedings of the ASME Dynamic Systems and Control Conference*, October 2008.

[33] J. F. Program and J. A. Ferwerda, "Three varieties of realism in computer graphics," *In Proceedings SPIE Human Vision and Electronic Imaging 2003*, vol. 5007, January 2003, pp. 290–297.

[34] G. S. Miller and C. R. Hoffman, "Illumination and reflection maps: simulated objects in simulated and real environments," *Course Notes for Advanced Computer Graphics Animation. Chicago, IL: ACM SIGGRAPH*, Jul 1984, pp. 1–12.

[35] K. J. Dana, B. van Ginneken, S. K. Nayar, and J. J. Koenderink, "Reflectance and texture of real-world surfaces," *ACM Trans. Graph.*, vol. 18, no. 1, pp. 1–34, 1999.

[36] J. R. Janesick, *Scientific Charge-Coupled Devices.* SPIE, Jan 2001.

[37] N. Khanna, A. K. Mikkilineni, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Scanner identification using sensor pattern noise," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1, San Jose, CA, USA, January 2007, p. 65051K.

[38] N. Khanna, A. K. Mikkilineni, P.-J. Chiang, M. V. Ortiz, S. Suh, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Sensor forensics: Printers, cameras and scanners, they never lie," *IEEE International Conference on Multimedia and Expo*, July 2007, pp. 20–23.

[39] N. Khanna, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Scanner identification with extension to forgery detection," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, no. 1, San Jose, CA, January 2008, p. 68190G.

[40] N. Khanna, A. K. Mikkilineni, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Survey of scanner and printer forensics at purdue university," *IWCF '08: Proceedings of the 2nd international workshop on Computational Forensics*. Berlin, Heidelberg: Springer-Verlag, August 2008, pp. 22–34.

[41] N. Khanna and E. J. Delp, "An overview of the use of distributed source coding in multimedia security (invited paper)," *WITMSE 2008, Workshop on Information Theoretic Methods in Science and Engineering*, Tampere, Finland, August 2008.

[42] N. Khanna, A. Roca, G. T. C. Chiu, J. P. Allebach, and E. J. Delp, "Improvements on image authentication and recovery using distributed source coding," vol. 7254, no. 1. SPIE, 2009, p. 725415.

[43] N. Khanna, A. K. Mikkilineni, and E. J. Delp, "Scanner identification using feature-based processing and analysis," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 123–139, March 2009.

[44] N. Khanna and E. J. Delp, "Forensic techniques for image source classification: A comparative study," *IWDW09: 8th International Workshop on Digital Watermarking*, Guildford, UK, August 2009, to appear.

[45] ——, "Source scanner identification for scanned documents," *First IEEE International Workshop on Information Forensics and Security*, London, UK, December 2009, submitted.

[46] I. Avcibas, N. D. Memon, M. Ramkumar, and B. Sankur, "A classifier design for detecting image manipulations," *Proceedings of the IEEE International Conference on Image Processing*, 2004, pp. 2645–2648.

[47] M.-J. Tsai and G.-H. Wu, "Using image features to identify camera sources," *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. II, Toulouse, France, May 2006, pp. 297–300.

[48] O. Çeliktutan, I. Avcibas, B. Sankur, and C. Capar, "Source cell-phone identification," *Signal Processing and Communications Applications, 2006 IEEE 14th*, April 2006, pp. 1–3.

[49] O. Çeliktutan, I. Avcibas, and B. Sankur, "Blind identification of cellular phone cameras," *Security, Steganography, and Watermarking of Multimedia Contents IX. Edited by Delp, Edward J., III; Wong, Ping Wah. Proceedings of the SPIE*, ser. Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, vol. 6505, Feb 2007, pp. 1 – 12.

[50] S. Bayram, H. T. Sencar, and N. Memon, "Improvements on source camera-model identification based on cfa interpolation," *Proc. of WG 11.9 Int. Conf. on Digital Forensics*, 2006.

[51] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[52] M. K. Mihcak, I. Kozintsev, K. Ramchandran, and P. Moulin, "Low-complexity image denoising based on statistical modeling of wavelet coefficients," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 300–303, 1999.

[53] J. Lukas, J. Fridrich, and M. Goljan, "Detecting digital image forgeries using sensor pattern noise," vol. 6072, no. 1. SPIE, 2006, p. 60720Y.

[54] Autodesk. (2001) Fake or foto? Http://area.autodesk.com/fakeorfoto.

[55] S. Lyu and H. Farid, "How realistic is photorealistic?" *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, Feb. 2005.

[56] T.-T. Ng, S.-F. Chang, J. Hsu, L. Xie, and M.-P. Tsui, "Physics-motivated features for distinguishing photographic images and computer graphics," *Proceedings of the 13th annual ACM international conference on Multimedia*. New York, NY, USA: ACM, November 2005, pp. 239–248.

[57] Y. Wang and P. Moulin, "On discrimination between photorealistic and photographic images," *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, May 2006, pp. 161–164.

[58] J. Wu, M. V. Kamath, and S. Poehlman, "Detecting differences between photographs and computer generated images," *Proceedings of the 24th International conference on Signal processing, pattern recognition, and applications*. Anaheim, CA, USA: ACTA Press, February 2006, pp. 268–273.

[59] W. Chen, Y. Shi, and G. Xuan, "Identifying computer graphics using hsv color model and statistical moments of characteristic functions," *IEEE International Conference on Multimedia and Expo*, July 2007, pp. 1123–1126.

[60] W. Chen, Y. Shi, G. Xuan, and W. Su, "Computer graphics identification using genetic algorithm," *19th International Conference on Pattern Recognition*, December 2008, pp. 1–4.

[61] F. Pan, J. Chen, and J. Huang, "Discriminating between photorealistic computer graphics and natural images using fractal geometry," *Science in China Series F: Information Sciences*, vol. 52, no. 2, pp. 329–337, February 2009.

[62] A. Dirik, S. Bayram, H. Sencar, and N. Memon, "New features to identify computer generated images," *IEEE International Conference on Image Processing*, vol. 4, October 2007, pp. 433–436.

[63] S. Dehnie, H. T. Sencar, and N. D. Memon, "Digital image forensics for identifying computer generated and digital camera images," *IEEE International Conference on Image Processing*, October 2006, pp. 2313–2316.

[64] A. Foi, V. Katkovnik, K. Egiazarian, and J. Astola, "A novel local polynomial estimator based on directional multiscale optimizations," *Proceedings of the 6th IMA Int. Conf. Math. in Signal Processing*, vol. 5685, no. 1, 2004, pp. 79–82.

[65] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Source digital camcorder identification using sensor photo-response non-uniformity," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1. SPIE, 2007, pp. 1G–1H.

[66] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2000.

[67] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[68] C.-W. Hsu, C.-C. Chang, and C.-J. Lin, "A practical guide to support vector classification." `http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf`

[69] A. C. Bovik, *Handbook of Image and Video Processing (Communications, Networking and Multimedia)*. Orlando, FL, USA: Academic Press, Inc., 2005.

[70] J. Fridrich, D. Soukal, and J. Lukáš, "Detection of copy-move forgery in digital images," *Proceedings of DFRWS 2003*, August 2003.

[71] A. Popescu and H. Farid, "Exposing digital forgeries by detecting duplicated image regions," Department of Computer Science, Dartmouth College, Tech. Rep. TR2004-515, 2004. `www.cs.dartmouth.edu/farid/publications/tr04.html`

[72] M. Chen, J. J. Fridrich, J. Lukás, and M. Goljan, "Imaging sensor noise as digital x-ray for revealing forgeries," *Lecture Notes in Computer Science-Information Hiding*, 2007, pp. 342–358.

[73] A. K. Mikkilineni, P.-J. Chiang, G. N. Ali, G. T.-C. Chiu, J. P. Allebach, and E. J. Delp, "Printer identification based on textural features," *Proceedings of the IS&T's NIP20: International Conference on Digital Printing Technologies*, vol. 20, Salt Lake City, UT, October/November 2004, pp. 306–311.

[74] (2009) Ocrad - the gnu ocr. Http://www.gnu.org/software/ocrad/.

[75] A. K. Mikkilineni, G. N. Ali, P.-J. Chiang, G. T. Chiu, J. P. Allebach, and E. J. Delp, "Signature-embedding in printed documents for security and forensic applications," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VI*, vol. 5306, San Jose, CA, January 2004, pp. 455–466.

[76] A. Vailaya, A. Jain, and H. J. Zhang, "On image classification: city vs. landscape," *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries*, vol. 0, June 1998, pp. 3–8.

[77] A. Payne and S. Singh, "Indoor vs. outdoor scene classification in digital photographs," *Journal of Pattern Recognition*, vol. 38, no. 10, pp. 1533 – 1545, April 2005.

[78] F. Cutzu, R. Hammoud, and A. Leykin, "Distinguishing paintings from photographs," *Journal of Computer Vision and Image Understanding*, vol. 100, no. 3, pp. 249–273, December 2005.

[79] V. Athitsos, M. Swain, and C. Frankel, "Distinguishing photographs and graphics on the world wide web," *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries*, vol. 0. Los Alamitos, CA, USA: IEEE Computer Society, June 1997, pp. 10–17.

[80] P. M. Rademacher, "Measuring the perceived visual realism of images," Ph.D. dissertation, 2003, director-Bishop, Gary.

[81] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, June 1998.

[82] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines (and other kernel-based learning methods)*. Cambridge University Press, 2000.

APPENDICES

A: SUPPORT VECTOR MACHINE (SVM)

Suppose we are given training data $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)$ where $y_i \in \{1, -1\}$. The vectors $\mathbf{x}_i, \forall i$ represent the feature vectors input to the SVM classifier and y_i represent the corresponding class labels. Assuming that the class represented by the subset $y_i = 1$ and the class represented by $y_i = -1$ are "linearly separable", the equation of a decision surface in the form of a hyperplane that does the separation is $\mathbf{w}^T\mathbf{x}+b = 0$; where, $\mathbf{x}$ is an input vector, $\mathbf{w}$ is an adjustable weight vector, and b is a bias.

For a given weight vector $\mathbf{w}$ and bias b, the separation between the hyperplane and the closest data point is known as the margin of separation, denoted by M. The goal of a support vector machine is to find the particular hyperplane for which the margin of separation M is maximized [68, 81, 82]. Under this condition the decision surface is referred to as the optimum separating hyperplane (OSH) ($\mathbf{w}_o^T\mathbf{x} + b_o = 0$).

The pair $(\mathbf{w}_o, b_o)$ with appropriate scaling, must satisfy the constraint:

$$\mathbf{w}_o^T\mathbf{x} + b_o \geq 1 \quad \forall y_i = +1 \tag{A.1}$$

$$\mathbf{w}_o^T\mathbf{x} + b_o \leq -1 \quad \forall y_i = -1 \tag{A.2}$$

The particular data points $(\mathbf{x}_i, y_i)$ for which $y_i[\mathbf{w}^T\mathbf{x}_i+b] = 1$ are known as support vectors, hence the name "Support Vector Machine." The support vectors are the data points that lie closest to the decision surface and are therefore the most difficult to classify. As such they have the direct bearing on the optimum location of the decision surface. Since the distance to the closest point is $\frac{1}{\|\mathbf{w}\|}$, finding the OSH amounts to minimizing $\| \mathbf{w} \|$ with the objective function: $min\ \phi(\mathbf{w}) = \frac{1}{2} \| \mathbf{w} \|^2$ subject to the constraints shown in Equations A.1 and A.2.

If $(\alpha_1, \alpha_2..., \alpha_N)$ are the N non-negative Lagrange multipliers associated with constraints in Equations A.1 and A.2, the OSH can be uniquely constructed by solving a constrained quadratic programming problem. The solution $\mathbf{w}$ has an expansion

$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$ in terms of a subset of training classes, known as support vectors, which lie on the margin. The classification function can thus be written as

$$f(\mathbf{x}) = \text{sgn}(\sum_i \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b) \tag{A.3}$$

If the data is not linearly separable, SVM introduces slack variables and a penalty factor such that the objective function can be modified as

$$\phi(\mathbf{w}) = \frac{1}{2} \parallel \mathbf{w} \parallel^2 + C(\sum_{i=1}^{N} \zeta_i) \tag{A.4}$$

Additionally, the input data can be mapped through some nonlinear mapping into a higher-dimensional feature space in which the optimal separating hyperplane is constructed. Thus the dot product required in Equation A.3 can be represented by $k(\mathbf{x}, \mathbf{y}) = (\phi(\mathbf{x}).\phi(\mathbf{y}))$, when the kernel k satisfy Mercer's condition [82]. Finally, the classification function is obtained as

$$f(\mathbf{x}) = \text{sgn}(\sum_i \alpha_i \mathbf{y}_i k(\mathbf{x}_i, \mathbf{x}) + b) \tag{A.5}$$

Because the SVM can be analyzed theoretically using concepts from statistical learning theory, it has particular advantage in problems with limited training samples in high-dimensional space.

B: GRAYLEVEL CO-OCCURRENCE MATRIX (GLCM) BASED FEATURES

This appendix describes the process of (GLCM) based feature extraction from a block of an image and specified dr and dc. These features model image texture [73]. Feature extraction is done by first determining the Graylevel Co-occurrence Matrix (GLCM), which is an estimate of the second order probability density function of the pixels in the image. The features are the statistics obtained from the GLCM and two pixel based features.

Figure B.1 shows an example block of a graylevel image, $Img(i, j)$, for which GLCM features are extracted. The region of interest (ROI) is the set of all pixels within the rectangular bounding box around the image.

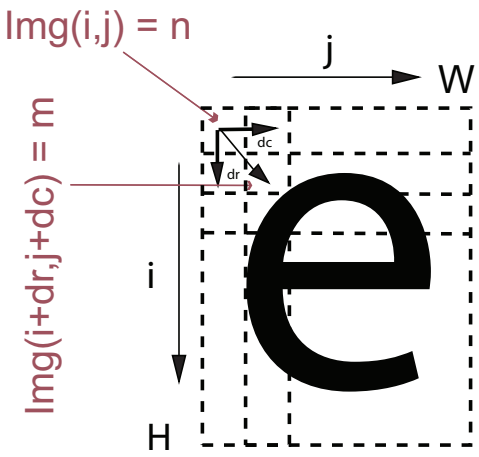


Fig. B.1. Example Image Block for Generation of GLCM Features.

The number of pixels R, in the ROI is defined as

$$R = \sum_{(i,j)\epsilon ROI} 1. \tag{B.1}$$

The Gray-Level Co-occurrence Matrix (GLCM), defined in Equation B.2, has entries $glcm(n, m, dr, dc)$ which are equal to the number of occurrences of pixels with graylevels n and m respectively with a separation of (dr, dc) pixels (Figure B.1). The

number of pixels for which this estimate is obtained is given by Equation B.3. If the GLCM is normalized such that its entries sum to one, the entries then represent the probability of occurrence of pixel pairs with graylevels n and m with separation (dr,dc).

$$glcm(n,m,dr,dc) = \sum_{(i,j),(i+dr,j+dc)\epsilon ROI} 1_{\{Img(i,j)=n, Img(i+dr,j+dc)=m\}} \tag{B.2}$$

$$R_{glcm} = \sum_{(i,j),(i+dr,j+dc)\epsilon ROI} 1 \tag{B.3}$$

$$p_{glcm}(n,m) = \frac{1}{R_{glcm}} glcm(n,m) \tag{B.4}$$

We now describe twenty features which are obtained from the GLCM. The first four are the marginal means and variances defined by Equations B.7-B.10 which are estimated from the marginal probability densities defined by Equations B.5 and B.6.

$$p_r(n) = \sum_{m=0}^{255} p_{glcm}(n,m) \tag{B.5}$$

$$p_c(n) = \sum_{m=0}^{255} p_{glcm}(n,m) \tag{B.6}$$

$$\mu_r = \sum_{n=0}^{255} p_r(n) \tag{B.7}$$

$$\mu_c = \sum_{m=0}^{255} p_c(m) \tag{B.8}$$

$$\sigma_r^2 = \sum_{n=0}^{255} n^2 p_r(n) - \mu_r^2 \tag{B.9}$$

$$\sigma_c^2 = \sum_{m=0}^{255} m^2 p_c(m) - \mu_c^2 \tag{B.10}$$

The next seven features are the energy of the normalized GLCM, three entropy measurements, the maximum entry in the GLCM, and two correlation metrics. These are defined by Equations B.11- B.17.

$$Energy = \sum_{n=0}^{255}\sum_{m=0}^{255} p_{glcm}^2(n,m) \tag{B.11}$$

$$hxy1 = -\sum_{n=0}^{255}\sum_{m=0}^{255} p_{glcm}(n,m)\log_2(p_r(n)p_c(m)) \tag{B.12}$$

$$hxy2 = -\sum_{n=0}^{255}\sum_{m=0}^{255} p_r(n)p_c(m)\log_2(p_r(n)p_c(m)) \tag{B.13}$$

$$h_{glcm} = -\sum_{n=0}^{255}\sum_{m=0}^{255} p_{glcm}(n,m)\log_2 p_{glcm}(n,m) \tag{B.14}$$

$$MaxProb = \max_{n,m}\{p_{glcm}(n,m)\} \tag{B.15}$$

$$\rho_{nm} = \sum_{n=0}^{255}\sum_{m=0}^{255} \frac{(n-\mu_r)(m-\mu c)p_{glcm}(n,m)}{\sigma_r\sigma_c} \tag{B.16}$$

$$diagcorr = \sum_{n=0}^{255}\sum_{m=0}^{255} |n-m|(n+m-\mu_r-\mu_c)p_{glcm}(n,m) \tag{B.17}$$

Four features, Equations B.19- B.22, are obtained from the difference histogram defined by Equation B.18. They are the energy, entropy, inertia, and local homogeneity of $D(k)$ respectively.

$$D(k) = \sum_{\substack{0\le n\le 255\\ 0\le m\le 255\\ |n-m|=k}} p_{glcm}(n,m) \tag{B.18}$$

$$Denergy = \sum_{k=0}^{255} D(k) \tag{B.19}$$

$$h_D = -\sum_{k=0}^{255} D(k)\log_2 D(k) \tag{B.20}$$

$$I_D = \sum_{k=0}^{255} k^2 D(k) \tag{B.21}$$

$$h_D = \sum_{k=0}^{255} \frac{D(k)}{1+k^2} \tag{B.22}$$

The last five features, Equations B.25- B.29, are obtained from the sum histogram defined by Equation B.23. They are the energy, entropy, variance, cluster shade, and cluster prominence of $S(k)$ respectively.

$$S(k) = \sum_{\substack{0 \le n \le 255 \\ 0 \le m \le 255 \\ n+m=k}} p_{glcm}(n,m) \tag{B.23}$$

$$\mu_S = \sum_{k=0}^{510} kS(k) \tag{B.24}$$

$$Senergy = \sum_{k=0}^{510} S(k) \tag{B.25}$$

$$h_S = -\sum_{k=0}^{510} S(k) \log_2 S(k) \tag{B.26}$$

$$\sigma_S^2 = \sum_{k=0}^{510} (k-\mu_S)^2 S(k) \tag{B.27}$$

$$A_D = \sum_{k=0}^{510} \frac{(k-\mu_r-\mu_c)^3 S(k)}{(\sigma_r^2+\sigma_c^2+2r\sigma_r\sigma_c)^{\frac{3}{2}}} \tag{B.28}$$

$$B_D = \sum_{k=0}^{510} \frac{k-\mu_r-\mu_c)^4 S(k)}{(\sigma_r^2+\sigma_c^2+2r\sigma_r\sigma_c)^2} \tag{B.29}$$

In addition to the twenty graylevel features above, two simple features are also included and defined in Equations B.31 and B.33. These are the variance and entropy of the pixel values in the ROI.

$$\mu_{Img} = \frac{1}{R} \sum_{(i,j)\epsilon ROI} Img(i,j) \tag{B.30}$$

$$\sigma_{Img}^2 = \frac{1}{R} \sum_{(i,j)\epsilon ROI} (Img(i,j)-\mu_{Img})^2 \tag{B.31}$$

$$p_{Img}(\alpha) = \frac{1}{R} \sum_{(i,j)\epsilon ROI} 1_{\{Img(i,j)=\alpha\}} \tag{B.32}$$

$$h_{Img} = -\sum_{\alpha=0}^{255} p_{Img}(\alpha) \log_2 p_{Img}(\alpha) \tag{B.33}$$

These twenty two features from anisotropic GLCM are extracted for each input image block for specified dr and dc.

VITA

VITA

Nitin Khanna was born in Varanasi, Uttar Pradesh, India. He earned his Bachelor of Technology degree in Electrical Engineering in 2005 from Indian Institute of Technology, Delhi, India where he graduated first in his department. His current research interests include image and video processing, multimedia forensics and information security.

Since 2005, Nitin has studied in the direct Ph.D. program at Purdue University in West Lafayette, Indiana. He has been a research assistant in the Video and Image Processing Laboratory (VIPER) under the supervision of Professor Edward Delp, the Charles William Harrison Distinguished Professor of Electrical and Computer Engineering. From 2006 - 2009, he was supported by a project funded by the National Science Foundation. During this time, he developed signal processing tools for image forensics. In the Spring 2007, Nitin was a member of a three person team that entered the CERIAS Research Poster Competition, winning first place.

Nitin Khanna is a student member of the IEEE.

Nitin Khanna's publications from this research work include:

Journal papers:

1. P.-J. Chiang, **Nitin Khanna**, Aravind K. Mikkilineni, Maria V. O. Segovia, Sungjoo Suh, Jan P. Allebach, George T.-C. Chiu, and Edward J. Delp, "Printer and scanner forensics," *IEEE Signal Processing Magazine*, vol. 26, no. 2, pp. 72–83, March 2009.
2. **Nitin Khanna**, Aravind K. Mikkilineni, and Edward J. Delp, "Scanner identification using feature-based processing and analysis," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 1, pp. 123–139, March 2009.
3. **Nitin Khanna**, Aravind K. Mikkilineni, and Edward J. Delp, "Forensic camera classification: Verification of sensor pattern noise approach," *Forensic Science Communications (FSC)*, vol. 11, no. 1, January 2009.
4. **Nitin Khanna**, Aravind K. Mikkilineni, Anthony F. Martone, Gazi N. Ali, George T.-C. Chiu, J. P. Allebach, and Edward J. Delp, "A survey of forensic characterization methods for physical devices," *Digital Investigation*, vol. 3, no. Supplement 1, pp. 17–28, July 2006.

Conference Papers:

1. **Nitin Khanna** and Edward J. Delp, "Source scanner identification for scanned documents," *First IEEE International Workshop on Information Forensics and Security*, London, UK, December 2009, submitted.
2. **Nitin Khanna** and Edward J. Delp, "Forensic techniques for image source classification: A comparative study," *IWDW09: 8th International Workshop on Digital Watermarking*, Guildford, UK, August 2009, to appear.
3. **Nitin Khanna**, Antoni Roca, George T. C. Chiu, Jan P. Allebach, and Edward J. Delp, "Improvements on image authentication and recovery using distributed source coding," vol. 7254, no. 1. SPIE, 2009, p. 725415.

4. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Survey of scanner and printer forensics at purdue university," *IWCF '08: Proceedings of the 2nd international workshop on Computational Forensics*. Berlin, Heidelberg: Springer-Verlag, August 2008, pp. 22–34.

5. **Nitin Khanna**, George T. Chiu, Jan P. Allebach, and Edward J. Delp, "Forensic techniques for classifying scanner, computer generated and digital camera images," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, NV, March 2008, pp. 1653–1656.

6. **Nitin Khanna**, George T. Chiu, Jan P. Allebach, and Edward J. Delp, "Scanner identification with extension to forgery detection," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819, no. 1, San Jose, CA, January 2008, p. 68190G.

7. **Nitin Khanna**, Aravind K. Mikkilineni, Pei J. Chiang, Maria V. Ortiz, Sungjoo Suh, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Sensor forensics: Printers, cameras and scanners, they never lie," *IEEE International Conference on Multimedia and Expo*, July 2007, pp. 20–23.

8. **Nitin Khanna**, Aravind K. Mikkilineni, Pei J. Chiang, Maria V. Ortiz, Vivek Shah, Sungjoo Suh, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Printer and sensor forensics," *IEEE Workshop on Signal Processing Applications for Public Security and Forensics (SAFE 07)*, April 2007, pp. 1–8.

9. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Forensic classification of imaging sensor types," *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX*, vol. 6505, no. 1. San Jose, CA, USA: SPIE, January 2007, p. 65050U.

10. **Nitin Khanna**, Aravind K. Mikkilineni, George T.-C. Chiu, Jan P. Allebach, and Edward J. Delp, "Scanner identification using sensor pattern noise," *Pro-*

ceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, no. 1, San Jose, CA, USA, January 2007, p. 65051K.