Indraprastha Institute of Information Technology Delhi

Doctoral Thesis

Estimation and Concealment of Forensic Multimedia Signatures

Author:
Ambuj Mehrish

Supervisor:
Dr. A.V Subramanyam

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Department of Electronics and Communication Engineering

December, 2019

©Indraprastha Institute of Information Technologyy (IIITD),
New Delhi 2019

Indraprastha Institute of Information Technology Delhi

Doctoral Thesis

Estimation and Concealment of Forensic Multimedia Signatures

Author:
Ambuj Mehrish

Supervisor:
Dr. A.V Subramanyam

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Department of Electronics and Communication Engineering

December, 2019

Certificate

This is to certify that the thesis titled, "Estimation and Concealment of Forensic Multimedia Signatures" being submitted by Ambuj Mehrish to the Indraprastha Institute of Information Technology Delhi, for the award of the degree of Doctor of Philosophy, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standard fulfilling the requirements of the regulations relating to the degree.

The results contained in this thesis have not been submitted in part or full to any other university or institute for the award of any degree or diploma.

Dr. A.V Subramanyam
Department of Electronics and Communication
Indraprastha Institute of Information Technology Delhi
New Delhi, 110020

"Sever the ignorant doubt in your heart with the sword of self-knowledge. Observe your discipline. Arise."

The Bhagavad Gita

Abstract

Ambuj Mehrish

Estimation and Concealment of Forensic Multimedia Signatures

The remarkable evolution of digital imaging techniques, processing and sharing in the past decades has spurred the penetration of multimedia into our lives. Unaccountable and ubiquitous use of multimedia brings severe issues and challenges about its origin and veracity. For instance, due to the growth of social media and instant messaging applications, the circulation of tampered content has become an unavoidable reality. The waning credibility of digital content has also lead to unfavorable consequences in terms of political, economic and social issues. Therefore, to address the source of origin and processing history-related issues of multimedia content, the scientific community has focused its attention on digital multimedia forensics techniques. Two of our major contributions belong to this class of forensics algorithms. On the other hand, to counter forensics algorithms, a parallel area of adversarial signal processing has also gained a lot of momentum. These counter-forensics algorithms are designed to hide the fingerprints left behind by image processing operations, thereby degrading the performance of forensics detectors. Our third major contribution belongs to this class of counter forensics techniques.

In our first contribution, we address the problem of establishing the link between a given image to its source camera device. This problem is called Source Camera Identification. In order to achieve this, the photo response non-uniformity (PRNU) characteristic of the sensor is exploited. However, the existing techniques suffer from the problem of noise induced during in-camera image processing. This noise can suppress the PRNU leading to poor camera identification performance. To this end, we propose a novel algorithm for robust estimation of PRNU from probabilistically obtained raw data. Since not all cameras provide raw data as their output, we compute raw data from the JPEG output using a probabilistic color de-rendering procedure. The estimated raw data is modeled as a Poisson process, and Maximum Likelihood Estimation is used for PRNU estimation. We also extend the estimated PRNU for tampering detection. The extensive experimental analysis performed on thousands of patches from various cameras reveal state-of-the-art performance. We also demonstrate the robustness of estimated PRNU by accurate tampering localization.

In our second contribution, we analyse the counter forensic algorithm for Contrast Enhancement, which is a common post-processing step in image tampering. The existing algorithms only consider the spatial domain. In our work, we consider changes in both spatial and DCT domain into account and obtain enhanced images such that the statistical properties are similar to natural images. Unlike the conventional techniques which lead to artifacts that can be captured by forensics detectors, the proposed algorithm suppresses the detectable artefacts. In our experiments, we demonstrate a significant performance degradation for deep learning as well as steganalysis-DCT feature based detectors. We also compute the popular image quality assessment metrics and show that the proposed model generates better visual quality images compared to the existing counter forensics techniques.

Nowadays cameras are also used in personal and commercial vehicles which pose a different problem of linking a given video to the vehicle in which the camera was mounted. This is useful for various applications, for example, insurance companies can authenticate the origin of video before processing the claim. In a different scenario of illegitimate video upload on the web, the video can be traced back to the car it originated from. To this end, we state our third contribution, in which we introduce the new area of multimedia vehicle forensics. We propose an algorithm for linking a dash-cam video to a specific car. Inspired by human gait bio-metrics, we observe that the subtle motion pattern of every vehicle can serve as its unique signature. We extract motion blur from dash-cam videos, which encode the motion pattern of the car. Experimental results on hours of dash-cam videos of several cars show the effectiveness of our approach. We further investigate the adversarial process of forging the signature of the vehicle and propose a forensics method to detect such forgery.

Acknowledgements

The work presented in this thesis would not have been possible without my close association with many people. I take this opportunity to extend my sincere gratitude and appreciation to all those who made this PhD thesis possible.

First and foremost, I would like to express my special appreciation and thanks to my advisor Dr. A.V Subramanyam, you have been a tremendous mentor for me. I would like to thank you for encouraging my research and for allowing me to grow as a researcher. Your guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my PhD study.

My sincere thanks also goes to Professor Mohan Kankanhalli for offering me the internship opportunity in his group. Working with a diverse group in his lab helped me improve my skills academically and professionally.

I must thank all the professors of IIIT-Delhi whom I have worked with over the last five years for showing me what it means to be a dedicated researcher, each in their own unique way. Each one of them have taught me that my job in life is to learn and explore new dimensions. I would especially like to thank my committee members for their valuable comments: Dr. Saket Anand, Dr. Sumit Darak and Dr. Min Wu. I am extremely grateful to them for their fruitful comments throughout my doctoral research endeavor. My gratitude also extends to the admin staff of IIIT-D, especially Ms. Priti for being extremely helpful in the fast resolution of all admin related matters.

My special gratitude to Parag Aggarwal, Dhananjay Kimothi and Lokendra Tiwari for being with me in thick's and thins of life, I find myself lucky to have friends like them in my life.

My parents and my sister Swati receive my deepest gratitude and love for their dedication and the many years of support during my graduate studies that provided the foundation for this work.

Last, but not least, I would like to thank my wife Reshu for her understanding and love during the final year of my PhD. Her support and encouragement made this dissertation possible.

Contents

Certificate **iii**

Abstract **vii**

Acknowledgements **ix**

Contents **xi**

List of Figures **xv**

List of Tables **xix**

List of Abbreviations **xxi**

List of Symbols **xxiii**

1 Introduction **1**
1.1 Introduction to Multimedia Forensics 1
1.1.1 Active and Passive Forensics 2
1.1.2 Counter Forensics 4
1.1.3 Multimedia vehicle forensics 4
1.2 Research Objectives 6
1.3 Contributions 7
1.3.1 Source Camera Identification and Tamper Detection . 7
1.3.2 Adaptive Contrast Enhancement Counter Forensics . . 8
1.3.3 Egocentric analysis of dash-cam videos 8
1.4 Structure of the Thesis 9

2 Literature Review **11**
2.0.1 Source Camera Identification and Tamper Detection . 11
2.0.2 Adaptive Contrast Enhancement-Based Counter Forensics 14
2.0.3 Egocentric analysis of dash-cam videos 16

3 Preliminaries **17**
3.1 Source Camera Identification 17
3.2 Features generated in different stages of image acquisition . . 18
3.2.1 Image acquisition pipeline 18
Optical imperfections 19
Color Filter Array (CFA) and Demosaicing 19

Senor Pattern Noise (SPN) 20
3.3 PRNU Estimation 20
3.3.1 Maximum Likelihood Estimation 20
3.4 Contrast Enhancement Counter Forensics 23
3.5 Egocentric analysis of dash-cam videos 25

4 Source Camera Identification and Tamper Detection 29
4.1 Proposed Algorithm 29
4.1.1 JPEG to raw conversion 29
4.1.2 PRNU estimation from raw data 32
4.1.3 Estimating true sensor data from noisy sensor data . . 33
4.1.4 Cramer Rao lower bound 34
4.1.5 Decision Statistics 35
4.1.6 Hypothesis Testing Formulation 36
4.2 Experiments Setup and Results 37
4.2.1 Experimental Setup 37
4.2.2 Classification Results 38
4.3 Forgery Detection 43

5 Adaptive Contrast Enhancement Counter Forensics 47
5.1 Natural Image Prior 47
5.2 Proposed Algorithm 48
5.2.1 Huber Markov Random Field Prior 48
5.2.2 Optimization Formulation 49
5.2.3 Solution to Optimization Formulation 50
5.3 Experimental Setup and Results 51
5.3.1 Parameters 52
5.3.2 Performance with Global contrast enhancement techniques 53
5.3.3 Database and Classifier 54
5.3.4 Performance against CE Detectors 55
First and Second order statistics detector 55
Steganalysis features based detectors 56
Performance against deep learning based forensic detector 57
5.3.5 Comparison with other counter forensic techniques . . 58
5.3.6 Visual quality assessment 61
5.4 Discussion 61

6 Egocentric analysis of Dashcam videos 63
6.1 Proposed Algorithm 63
6.1.1 Blur extraction algorithm 63
6.1.2 Feature Extraction 64
6.1.3 Classifier 65
Decision Tree 66
Random Forest 66
AdaBoosting 67
6.1.4 Labeling Strategy 67

6.1.5 Random Forest Similarity . . . 68
6.2 Dataset . . . 69
6.2.1 Training and Testing Set . . . 71
6.3 Experiments Setup and Results . . . 71
6.3.1 Experimental Set Up . . . 71
6.3.2 Classification Results . . . 73
6.3.3 Using random forest similarity for prediction . . . 78
6.3.4 Forging Signature . . . 78
6.4 Application in Person Identification . . . 80
6.5 Discussion . . . 85

7 Conclusion and Future Directions **87**
7.1 Conclusion . . . 87
7.2 Future Directions . . . 88

8 Author's Publications **93**
8.1 Journal . . . 93
8.2 Conference . . . 93

A RAW to MATLAB **95**

B Blur Extraction **97**

C Parameter Estimation: Generalized Gaussian Distribution **101**
C.1 Mathematical Model . . . 101
C.2 Maximum Likelihood Estimation . . . 101

List of Figures

1.1 Facebook and Twitter were dominated with images of flooding and destruction caused by Hurricane Sandy in New York City. However, some images are photoshopped or directly taken from Hollywood movie[1]. 2
1.2 The Swiss tabloid Blick digitally altered a puddle of water (left) to appear as blood flowing (right) from the temple in order to magnify the amplitude of the tragedy. 3

2.1 Original image (Left) and noise residue obtained after subtracting denoised image from original image. Image content is clearly seen in noise residue which can alter the correlation between reference PRNU and noise residue. 12
2.2 (Left to right) GLCM of Original image, enhanced images using gamma correction ($\gamma = 0.6$), histogram stretching and enhanced image using CLAHE (contrast enhanced limit 0.01) . 15

3.1 Color image pipeline. Typical components include lense, image sensor, Bayer filter, image scaling, gamma correction, image enhancement, color space conversion (between formats such as RGB, YUV or YCbCr), image compression (such as JPEG) 18
3.2 Radial distortion in the rectangular grid. 18
3.3 The Bayer color filter mosaic. Each two-by-two sub-mosaic contains 2 green, 1 blue, and 1 red filter, each filter covering one pixel sensor. 19
3.4 The PRNU pattern is unique and spatially related to each region in an image. It is present in all pristine images produced by the camera but absent in tampered areas. By detecting the presence/absence of the camera PRNU in the image under test, a decision of forgery can be obtained. 22
3.5 (Left to right) Original image, enhanced image using gamma correction ($\gamma = 0.6$), and antiforensically enhanced image generated using [22] along with the respective histograms. Peaks are visible in the histogram of gamma corrected image (peaks are marked in red), whereas, the histogram of the remapped image is free of peaks. 24

[1] **http://pth.izitru.com/2012_10_02.html**.

3.6 (Left to right) Original image, enhanced image using gamma correction ($\gamma = 0.6$), and antiforensically enhanced [27] (AntiCE(GD)), [151] (AntiCE(TV)) along with the respective histogram in the bottom row. Histogram of gamma corrected image shows peaks and gaps (peaks are marked in red), whereas, histograms of antiforensically generated images are free of peaks. 25

4.1 JPEG images (first row) with corresponding estimated RAW images (second row). 30
4.2 Output image at each step of camera's image processing pipeline (Samsung NX2000). (a) Raw image with value between dark level and saturation level (b) linearized raw data $m \times n$ array with real values between 0 and 1 (c) $m \times n \times 3$ Demosaiced image (d) sRGB color space image, (e) Brightened sRGB image (f) Gamma corrected image (g) JPEG compressed image with Quality factor of 90. 30
4.3 First row: TIFF images obtained using dcraw [170], Second row: raw images after processing TIFF images and Third row: estimated raw images using [32] 31
4.4 Classification accuracy per camera for patch size 512×512 and 256×256 38
4.5 Classification accuracy per camera for patch size 512×512 (Left) and 256×256 (Right) with quality factor of 40 and 10 respectively. 40
4.6 Spatial domain (first three columns) and frequency domain (last three columns) representation of PRNU estimated by proposed (first row), MLE-PRNU (second row) and Phase-PRNU (third row) respectively for 3 different cameras 40
4.7 ROC plots for detection of synthetic forgeries with the 128×128 size window. 42
4.8 Error rate plots for Cannon 60oD. MLE-SPN, Phase-SPN and Proposed algorithm (left to right) respectively. 44
4.9 Localizing forgeries on realistic images using MLE-PRNU (4^{th} and 5^{th} column), Phase-PRNU (6^{th} and 7^{th} column), proposed (8^{th} and 9^{th} column). 1^{st} column shows original images, 2^{nd} column tampered images, 3^{rd} column ground truth respectively. Whereas green color indicates false positives and red true negative. 45

5.1 First row (left to right): Original image, enhanced images using the proposed algorithm for $\Gamma = \{0.08, 0.8, 1, 2\}$ respectively. Second Row (left to right): Original image, enhanced image for $\delta = \{0.5, 1, 5, 10\}$ respectively. 51

5.2 Input to the neural network proposed in [171] is the grayscale GLCM images of size 256×256. (Top row, left to right) Original image, Gamma Correction (GC), Histogram Stretching (HS), CLAHE. (Bottom row, left to right) Gaussian Dithering (GD), Total Variation (TV) and Proposed. 55
5.3 Performance analysis of forensic detectors based on SRM, SPAM and SPAM-CCPEV feature respectively. 58
5.4 Examples of images enhanced using CLAHE and images generated anti forensically. (Left to right) Original image, CLAHE enhanced image, proposed and [27]. In magnified portions of enhanced images generated using [27], we can observe visible noise whereas images generated using proposed algorithm are smooth and similar to conventionally enhanced images. . 59
5.5 Examples of images enhanced using CLAHE and images generated anti forensically. (Left to right) Original image, CLAHE enhanced image, proposed, [11] and [27]. 60

6.1 Figure explains feature extraction process from blur sequence for two consecutive windows. Window of size W is shifted by s and DFT is performed along the rows of the extracted window. $0, 1, 2....W$ denote frequency indices. The feature vector is created by stacking frequency components as shown in figure. (Best viewed in color) 65
6.2 0^{th} (top), 1^{st} (middle) and 2^{nd} (bottom) frequency feature plot using t-sne for 4 different cars. We can observe the clustering behavior for all three frequency components from same car. Also features from different cars are separated by considerable margin (Please refer to web version for better visualization). 66
6.3 Similarity calculation using random forest. Numbers in green represents same terminal node position whereas red signifies different. Out of 12 positions, 8 are similar where as 4 are different. Therefore S is equal to 2 using Eq. 6.3. 69
6.4 Database. 70
6.5 Classification accuracy for three different thresholds (in %) . 72
6.6 Blur kernel extracted from 6 consecutive frames for Honda city (top row), XUV (middle row) and Swift (last row) (Best viewed in color). 73
6.7 CMC curve for average accuracy over 10 fold cross validation. 74
6.8 ROC plot for different window size and shift. Window size = 5 and shift = 2 has better performance. Therefore, $W = 5$ and $s = 2$ are used for creating feature vectors in all experiments. 76
6.9 ROC plot for varying frequency components in feature vectors. 76

6.10 Similarity score for 15 different people computed using Eq. 6.3. Test and reference clips have same number of frames (1000 frames each). Higher value represents higher similarity between reference clip and test clip. Each cell represents similarity score of a video (in vertical axis) with reference video of corresponding person (in horizontal axis). 83

6.11 Histogram plot of the number of feature vectors terminating at the same node position in each decision tree. Blue histogram represent test and reference video belongs to same person, whereas the histogram in red represents every other case. Figure provides comparison of test video of Person1 with reference videos of every other person in the dataset. We can observe that when clip belongs to same person, many decision trees have feature vectors terminating at same node position, however in other case only few decision trees have feature vectors terminating at same node position. Such information can be used to differentiate between two clips, since the distribution is different for the two cases. (Please refer to the web version for better visualization. Decision trees on x-axis and Frequency on y-axis) 84

7.1 The pipeline for PRNU based image clustering with deep learning. 91

List of Tables

4.1 Details of Cameras Used for Experiments 37
4.2 38
4.3 Classification accuracy per camera for patch size 128×128 . . 39
4.4 Classification accuracy per camera for patch size 64×64 . . . 41
4.5 Model of cameras used for testing forgery localization and number of images used per camera for estimation of PRNU and Predictor. 43

5.1 Visual quality assessment between antiforensically enhanced images generated using global enhancement techniques such as GC, HS, and CLAHE, where a lower value indicates less distortion. Best results are underlined. 53
5.2 Performance analysis for 1^{st} and 2^{nd} order statistics based detectors (FPR = 0.01). 53
5.3 Performance analysis (accuracy in %) for steganalysis features based detectors (FPR = 0.01). CE is performed using CLAHE. 54
5.4 Performance analysis for steganalysis features based detectors for CLAHE using Rayleigh and Exponential distribution for enhancing images. $\alpha = 0.2$ is used for generating AntiCE images using Eq 5.5. 56
5.5 Performance (accuracy in %) against deep learning based forensic detectors. 57
5.6 Visual quality assessment between antiforensically enhanced images generated using [11], [27], [151] and proposed algorithm, where a lower value indicates less distortion [109]. Best results are underlined. 61

6.1 List of mobile phones used for capturing data. Apple iphone 6s and 5s have inbuilt video stabilization. 70
6.2 Performance of different classifier. Random forest (RF) performed better than other classifiers. AdaBoosting further improved the performance by approximately 2%. 74
6.3 Rank 1-5 accuracy for 25 cars (in %) and classification accuracy for three different thresholds. (in %) 75
6.4 Accuracy of the proposed algorithm (in %) under different conditions using RF-Adaboot classifier with majority voting. (*A:Same Road, B:Same Camera, C:Same scene content, D:Same light Condition)* 77

6.5 Similarity score S and ratio r computed using Eq. 6.3 and Eq. 6.4 respectively. '-' signifies ratio of 1. Numbers in bold represents maximum value of S and values of r less than $\tau = 2$ are marked in red. Possible predicted labels are computed by comparing value of r with τ, label of corresponding class is selected if ratio $r < \tau$. Similar results can be computed for other clips. (*Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio*) 79

6.6 Predicting possible labels for forged videos using similarity score S and ratio r computed using Eq. 6.3 and Eq. 6.4.'-' signifies ratio of 1, corresponding maximum similarity scores are in bold and values of r less than $\tau = 2$ are marked in red. Possible predicted labels are computed by comparing value of r with τ, label of corresponding class is selected if ratio $r < \tau$. Similar results can be computed for other clips. (*Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio*) 81

6.7 Table computes ratio r using similarity score S for video clips of 4 different persons. '-' signifies ratio of 1. Numbers in bold represents maximum similarity and values of r less than $\tau = 2$ are marked in red. Possible predicted labels are computed by comparing value of r with τ, label of corresponding class is selected if ratio $r < \tau$. (*Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio*) 82

List of Abbreviations

SCI	**S**ource **C**amera **I**dentification
CFA	**C**olor **F**ilter **A**rray
PRNU	**P**hoto **R**esponse **N**on **U**niformity
GLCM	**G**ray **L**evel **C**o-occurrence **M**atrix
WGN	**W**hite **G**aussian **N**oise
GGD	**G**eneralized **G**aussian **D**istribution
CLAHE	**C**ontrast **L**imited **A**daptive **H**istogram **E**qualization
GC	**G**amma **C**orrection
HS	**H**istogram **S**tretching
MRF	**M**arkov **R**andom **F**ield
HMRF	**H**uber **M**arkov **R**andom **F**ield
MSE	**M**ean **S**quare **E**rror
CRLB	**C**ramer **R**ao **L**ower **B**ound
MVU	**M**inimum **V**ariance **U**nbaised
MLE	**M**aximum **L**ikelihood **E**stimation
PCE	**P**eak **C**orrelation **E**nergy
CCN	**C**ross-**C**orrelation **N**orm
ZM	**Z**ero **M**ean
WF	**W**einer **F**ilter
FPR	**F**alse **P**ositive **R**ate
EER	**E**qual **E**rror **R**ate
TV	**T**otal **V**ariation
DCT	**D**iscrete **T**ime **T**ransform
HVS	**H**uman **V**isual **S**ystem
MAD	**M**ost **A**pparent **D**istortion
VIF	**V**isual **I**nformation **F**idelity
SSIM	**S**tructural **S**imilarity **I**ndex
PSNR	**P**eak **S**ignal to **N**oise **R**atio
STFT	**S**hort **T**ime **F**ourier **T**ransform

List of Symbols

Symbol	Description	Dimension
I, **X**, **K**	Vectors are represented by capital bold letters	$\mathbf{R}^n$
I, X, K	Matrices are represented by capital letters	$\mathbf{R}^{m \times n}$
c, x, g, k	Scalars and constants are represented by lower case letters	**R**

1 Introduction

1.1 Introduction to Multimedia Forensics

A picture can tell a thousand words. This is one of the reasons why the digital image is one of the most commonly used multimedia types on the Internet. Over the past decade, digital images have become a popular way to communicate, store, and process information. The constant advancement and convenience of image acquisition has eased the distribution and sharing of digital images. Further, powerful image editing software allows even amateurs to create forgeries for malicious or criminal use. As a result, we are rapidly reaching towards a situation where one can no longer take the authenticity and integrity of digital images for granted. This trend undermines the credibility of digital images presented as evidence in a court of law, as news items, as part of a medical record in scientific articles or as financial documents since it may no longer be possible to distinguish whether a given digital image is the original or a (maliciously) modified version or even a depiction of a real-life occurrences and objects. For example, Fig 1.1 shows some images circulated over the social media (Facebook and Twitter) during the landfall of Hurricane Sandy in New York City. Many images show the fake presence of shark in flooded areas and depict the images from Hollywood movie as images of the hurricane.

The incredibility associated with digital images has encouraged the development of digital multimedia forensics techniques. The main objectives of forensics are i) to analyze whether a given content is tampered; ii) to determine the origin; iii) to trace the processing history; iv) to reveal latent details invisible to naked human eyes.

Multimedia forensics involves the study and development of algorithms without relying on any information besides the digital content itself. This is possible because most of the signal processing operations performed on the content leave certain artifacts. These artifacts act as the fingerprint for the respective operation. Since these fingerprints are unique to a particular operation, it is hard to design a universal forensics method which can capture all the operations. Instead, different forensics techniques are designed to identify the fingerprints from various signal processing operations.

Though existing digital forensics techniques are capable of detecting several standard digital multimedia manipulations, they may not account for the possibility that anti-forensics operations designed to hide traces of manipulation may be applied to digital content. In practice, it is also possible that

a forger with a digital signal processing background may be able to secretly develop anti-forensics operations and use them to create undetectable digital forgeries. As a result, several multimedia forensics techniques may possess vulnerabilities that are unknown to the forensics community at large.

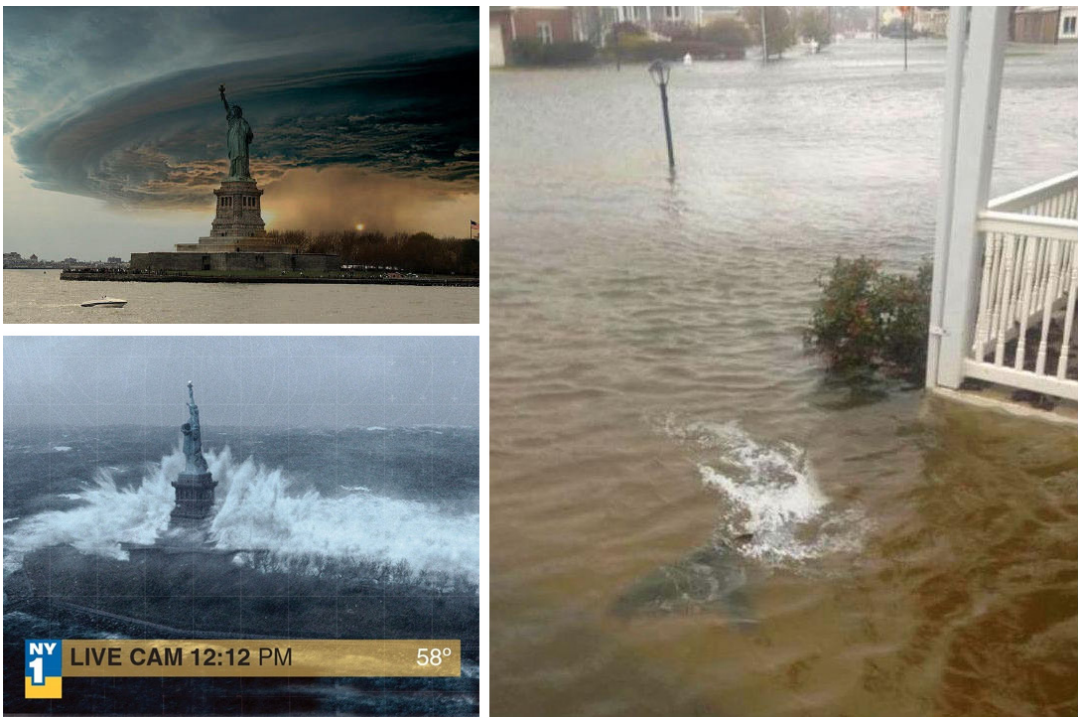


Figure 1.1: ***Facebook and Twitter were dominated with images of flooding and destruction caused by Hurricane Sandy in New York City. However, some images are photoshopped or directly taken from Hollywood movie[1].***

1.1.1 Active and Passive Forensics

Multimedia forensics algorithms are broadly classified under two categories "active forensics" and "passive forensics". Active forensics algorithms such as digital watermarking have been proposed for authentication of digital content. In active forensics approach, a watermark is embedded into a multimedia content before distributing. The watermark can be extracted at a later stage to establish authenticity, tamper localization, copyright violation detection, and ownership proving. The main drawback of the active forensics is that a specific signature or watermark is required to be embedded in digital content at the time of creation. This limits the application of such methods to specially equipped cameras or subsequent processing of the content.

On the other hand, passive techniques do not take any cognizance of the content creation process. The algorithms assume that the original or processed image have inherent artifacts which can be exploited to achieve multimedia forensics goals. In this thesis, we only consider passive forensics

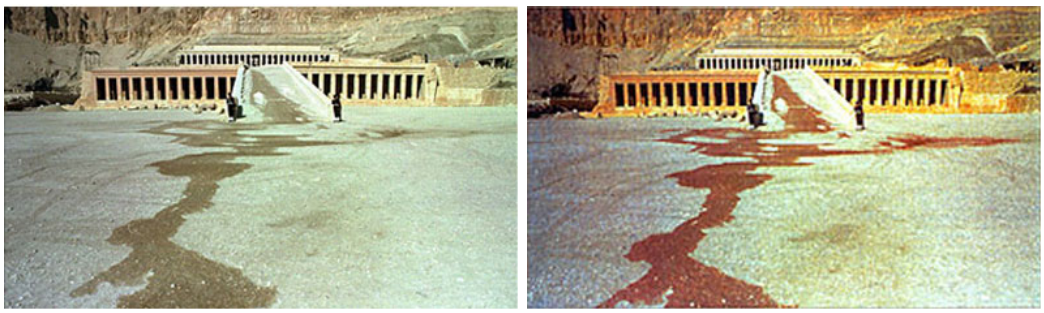


Figure 1.2: ***The Swiss tabloid Blick digitally altered a puddle of water (left) to appear as blood flowing (right) from the temple in order to magnify the amplitude of the tragedy.***

approach to address source identification and anti-forensics enhancement problem for images and videos.

Passive multimedia forensics can be broadly divided into the following three categories.

- Source camera identification
- Tampering detection and localization
- Image manipulation detection

Source camera identification ([16], [175], [107], [75], [5], [165], [176], [91], [180], [99], [30], [187], [3], [7], [34], [93], [69], [77], [114], [126], [186]) aims to authenticate the digital image by establishing a link between an image under investigation and the camera which was used to capture it. The device can be a camera, scanner, or mobile phone. The basic assumption is that digital pictures taken by the same device are overlaid by a specific pattern, that is a unique and intrinsic fingerprint of the acquisition device. We will discuss the intrinsic fingerprints in Chapter 3.

The second class of algorithms aims at detecting and localizing tampering ([39], [45], [119], [92], [114], [115], [85], [104]) in the content. In general, when an image is forged, the visual artifacts are not prominent, and it is hard to disclose the tampering at simple visual inspection. However, forgery disregards the natural statistics of an original untampered image, and these statistics can be exploited to detect and localize forgery. We discuss this in Chapter 4.

In the third class of passive forensics, algorithm aims to detect any enhancement or post-processing operation applied over the image ([153], [48], [166], [168], [169]). Post-processing operations such as filtering, enhancement, or compression are applied after the image is tampered. This is done to suppress the tampering artifacts as well as make the potentially tampered image perceptually more convincing. Median filtering and contrast enhancement are the two most frequently used image enhancement operations. Fig 1.2 shows that the representation of an image can be maliciously changed by performing contrast enhancement.

1.1.2 Counter Forensics

With increasing awareness about forensic techniques and capabilities, adversaries are making use of counter-forensics techniques to destroy or hide traces of manipulation in the image [13], [17], [19], [47], [76]. This makes forensic investigator's task even more difficult. The design of counter-forensic techniques is not only of interest to the adversary, but also allows to analyse the robustness of forensics algorithms.

The robustness of the forensic algorithm is defined as its ability to produce reliable results after post-processing (such as JPEG compression, median filtering, contrast enhancement). For example, if the quality of the image is reduced by lossy compression or resampling, an adversary can eliminate the identifying traces [62], [63], [64], [167]. In such scenarios, forensic algorithms which are not sensitive to strong quantization will produce more reliable results. As a consequence, during a forensic investigation, the robustness of forensic algorithms is of paramount importance. Thus, as part of our analysis of forensic algorithms, we also explore counter forensics for image enhancement.

1.1.3 Multimedia vehicle forensics

The conventional image and video forensics is primarily focused on captured content or source of acquisition. However, the rapid development in capturing modes throw different challenges. In this thesis, we introduce one such challenge.

Video surveillance with advanced video analytic algorithms [196], [36] can serve a variety of applications such as detecting crime, traffic, and crowd control management. However, video analysis from fixed overhead cameras mainly focus on techniques such as background modeling and detecting moving objects [71], [21], [118]. Due to advancement in technology, low cost of wearable visual sensors, and wide availability of cameras mounted on the ground as well as aerial vehicles, visual sensing based on moving cameras is gaining popularity. In one such example, police vehicles often use the camera mounted in their vehicle popularly known as dash-cams for license plate recognition. To improve evidence documentation, increased accountability and transparency, police officers have started to use body-wear cameras in patrolling operations [54]. Further, drones equipped with license plate scanning and thermal imaging as well as radio equipment along with several other sensors have become popular in a variety of applications [61], [164], [29] especially in law enforcement. There is a significant scope of expansion in traditional video surveillance due to the development of these mobile devices by providing quicker and effective methods for preventing crime. Such devices also find use in surveillance for civil security in large gatherings [52], [55], [58].

An unusual account of dash-cam popularity is the massive recording of a meteor shower dazzling in the sky over Chelyabinsk, Russia in 2013 [57]. This phenomenon was possible because most of the cars in Russia are equipped with dash-cams. Dash-Cams are becoming rapidly popular in many countries around the world. These moving sensors generate a rich amount of video content. The web is flooded with thousands of such videos showing road accidents, bad driving behavior, and frightening close-shaves. It will not be a surprise to witness dash-cams becoming a standard feature of not only cars but also other motor vehicles such as trucks and vans due to the falling price of cameras.

In [141], Park *et al.* conducted survey on motives and concerns people have while sharing dashcam videos. Their survey shows that dashcams are mainly used to collect evidence for traffic-related accidents/incidents, since, in recent years, there has been a rise in fraudulent claims. Also, In recent years, Youtube is infused with recordings of everything from meteors to crashes [57] and lucky escapes [53]. However, in some countries, the people sharing (via publication) the dash-cam footage could face legal action [60] for violating the privacy of other road users. A lot of sensitive information about other cars is recorded by dash-cams such as the car number plate, and this information can be used by various agencies for their benefit. Linking the video to a particular vehicle can be useful in investigating many such cases.

Dash-cams enable recording of video footage that can be used to track down law offenders, examine details of accidents, and incidents of misconduct on roads. There is no legal obligation to act as a witness to such events. However, such videos are potentially admissible as evidence in a court of law. Therefore, for preventing a false testimony, authenticating the source of the video is required before presenting it as evidence. It is again useful to link the video with the witness car which can provide us with the vital information that the witness's car was present where the crime took place, and thus making the witness's claim even stronger and protecting the driver's privacy by not divulging the witness driver's details.

Vehicle identification is a well-studied research problem [95], [163], [162]. The previous work done towards identifying the vehicle make use of expensive equipment such as laser vibrometer. A laser doppler vibrometer (LDV) is an optical instrument that quantifies mechanical oscillations. It finds its application in the automotive industry for vibration analysis of individual components and the entire vehicle. However, we propose to use dashcam videos for measuring the vibration signature of the car because dashcams are more popular and accessible to the general public when compared to LDV. Since dashcam is receiving the same mechanical vibration, which LDV can measure from the surface of the vehicle using laser, the underlying pattern extracted by dashcam and LDV will be similar. Extracting this signal from dashcam is ill-posed problem because we only observe the blurred frame. The blur kernel and sharp unblurred frame are unavailable, making the estimation problem under-determined. Approaching vehicle forensics

in this way is different from traditional forensics problems. As in conventional forensic problems, we can directly estimate the signal of interest from the images or videos generated using the source. For example, in PRNU estimation nature of the sensor pattern noise is known whereas in blur estimation, the nature of blur is unknown. In another such example, while performing contrast enhancement forensics, the analyst is aware of the type of unique signatures or artifacts to search in images for performing analysis, whereas while extracting a signature from dashcam videos, type of blur introduced by the motion of the vehicle is not known. This approach is similar to egocentric video analysis proposed in [65], [66], [67], [83].

1.2 Research Objectives

The major problems in multimedia forensics are source identification, forgery detection and localization, and image manipulation detection [16], [175], [107], [75], [5], [165], [176], [91], [180], [99], [30], [187][3], [7], [34], [93], [69], [77], [114], [126], [186]. In this work, we primarily focus our attention on forensic [166], [168], [169], [25], [28] and counter forensic problems [26], [108], [41] related to the following areas.

- **Source Camera Identification and Tamper Detection**
  - We propose a robust technique for estimation of the sensor pattern noise which is popularly used in camera identification and extend our approach for detecting the forgery in images. We further use estimated sensor pattern noise for forgery localization, and with extensive experiments, we show that forgery localization is improved by using the proposed estimation method.
- **Adaptive Contrast Enhancement-Based Counter Forensics**
  - Adaptive contrast enhancement improves the local contrast of an image. Since images are usually modified locally, adaptive enhancement is useful in hiding the traces of tampering. We propose a counter forensics algorithm which enhances the image without being recognized by forensic detectors. Thus by concealing the traces of enhancement, we highlight the gaps in current forensics models which adversary can exploit.
- **Egocentric analysis of dash-cam videos**
  - In addition to the classical forensics problems, we introduce the emerging area of multimedia vehicle forensics. The ever-changing ways of capturing the content also opens up new issues. In one such specific case, vehicle dashcams have become a popular way of recording the videos. This acquisition mode poses significant source linkage issues of the given video with dashcam vehicle as well as privacy challenges. We propose an algorithm for linking

a dash-cam video to a specific car. We further investigate the process of forging the signature of a car and propose a counter forensics method to detect such forgery. Also, we discuss the application of our technique to other potential platforms where the camera can be mounted, for example, on the chest of a person.

In the following section, we briefly discuss our contribution in respective areas.

1.3 Contributions

In this dissertation, we propose algorithms for estimating the signatures from multimedia content. Extracted signatures can be used in forensic applications; for example probabilistically computed photo response non uniformity (PRNU) is used for camera identification, tamper detection and localization. Further, for identifying the gaps in forensic algorithms, we propose an optimization framework for generating contrast-enhanced images which are robust to forensics detection models. We also propose a novel method for vehicle forensics using multimedia signatures. The proposed algorithm can be used for linking the dash-cam video with its source, i.e., with the vehicle in which dash-cam is installed. Detailed contributions are as follows.

1.3.1 Source Camera Identification and Tamper Detection

- Our major contributions are twofold. First, we propose a novel algorithm for extracting PRNU from probabilistic raw values. Second, we perform an extensive analysis of the algorithm for forgery localization using segmentation guided analysis [104].
- We compare our results with popular estimation algorithms (MLE-PRNU [34] and Phase-PRNU [92]) and demonstrated mathematically that the estimate of PRNU is minimum variance unbiased (MVU) and is consistent with CRLB bound.
- We perform elaborate experiments for patch of different sizes (64×64 to 512×512) and present an extensive evaluation of the proposed algorithm for tamper localization.
- We evaluate proposed algorithm on synthetic and realistic forgeries of size 64×64 and 128×128.

1.3.2 Adaptive Contrast Enhancement Counter Forensics

- We consider artifacts in both spatial and DCT domain and obtain enhanced images similar to natural images in statistical properties. We present a novel optimization formulation, and analytically show the solution steps.
- We use adaptive contrast enhancement (CLAHE) as enhancement operation in comparison to previous works, which only deals with global contrast enhancement. For generating adaptive contrast enhanced images, we use Uniform, Rayleigh, and Exponential distribution [197] as the basis for creating the contrast transform function.
- We test our proposed algorithm on the first order [28], second-order [48], and deep learning-based state-of-art detectors [171], [160]. Further, we compare the performance of the proposed algorithm with popular antiforensic algorithms [11], [27], [151], and establish the superior performance obtained by our model. Additionally, we perform a rigorous image quality assessment of the antiforensic images and show that the proposed algorithm achieves better image quality.

1.3.3 Egocentric analysis of dash-cam videos

- We propose a novel method for linking the video recorded with a dash-cam to the vehicle on which it is mounted. We emphasize and discuss the need for vehicle forensics due to the increase in the use of mounted cameras in vehicles.
- We propose to extract a unique signature from the blur kernels of dash-cam videos which are embedded due to the subtle motion of a vehicle.
- The hypothesis that blur kernels arising due to motion pattern can lead to the identification of the source in which the camera is mounted is not limited to vehicles only. It can be generalized to other potential platforms, for example, a camera mounted on the chest of a human. To this end, we collect a novel dataset and show that the algorithm is applicable in this scenario.
- We also introduce adversarial processing and show that signatures used in a forensic investigation can be forged to mislead the investigation. We propose a possible solution for the case when a video is forged with the signature of other vehicles. We use robust classification algorithm (random forest) and a metric (Cumulative match curve) for the representation of results.
- Also, we discuss that the new area of multimedia vehicle forensics is opened up by this work and has many exciting new research problems and needs urgent solutions.

1.4 Structure of the Thesis

This thesis is organized as follows.

In Chapter 2, we present the literature review.

Chapter 3 gives background knowledge on source camera identification, image forgery detection, forensics and counter forensics of contrast enhancement and egocentric video analysis. In this chapter, we review state of the art PRNU estimation methods and contrast enhancement forensics/counter forensics algorithms. We also discuss evaluation metrics, natural image datasets used in forensics testing of this thesis, and relevant optimization methods which will be used in the proposed algorithms.

In Chapter 4, we propose an algorithm for the estimation of PRNU from RAW data, which is obtained from JPEG compressed images. The RAW data is obtained from JPEG images using a probabilistic approach. We then use a joint Poisson probability model to define the distribution of raw values obtained from several images. To estimate the PRNU component of the camera, we use MLE and the true value of raw data obtained by denoising probabilistically generated raw data. We also present results of experiments on a wide gamut of images from a composite database and evaluate its performance against other popular camera identification algorithms discussed in Chapter 3.

Chapter 5 describes a novel optimization formulation which enhances the attacked image without being detected by CE detectors. Our approach is the first CE counter forensics method, which works jointly on spatial and frequency domain. We use adaptive contrast enhancement (CLAHE) as enhancement operation in comparison to previous works, which only deals with global contrast enhancement.

In Chapter 6, we propose the novel problem of linking the video recorded with dash-cam with the vehicle on which it is mounted. We introduce forensics and counter forensics for vehicle forensics and also extend our proposed algorithm for the person identification problem.

Chapter 7 concludes this thesis by summarizing the contributions and proposing several perspectives about future research work on multimedia forensics.

2 Literature Review

There has been a great interest to study the features that can be used for source identification and enhancement detection. The objective of this chapter is to provide an overview of the available literature. First in Section 2.0.1, we briefly discuss camera identification techniques. In Section 2.0.2, we discuss forensic and counter forensic algorithms used for detecting and hiding the image enhancement. Since the egocentric analysis of dash cam video is a relatively new area, in Section 2.0.3, we discuss closely related work which deals with the egocentric analysis of multimedia content.

2.0.1 Source Camera Identification and Tamper Detection

Source camera identification (SCI) and image forgery detection are two of the key problems in multimedia forensics. The source camera identification aims to verify whether a given digital image is acquired by a particular device or an individual camera model. The image forgery detection seeks to detect any act of manipulation such as splicing, cloning or deletion in an image.

Image acquisition pipeline leaves various artifacts in images such as (CFA) interpolation artifacts [16], color filter array [175], defective pixels [107], [75], JPEG compression artifacts [5], [165], [176], lens aberration [91], [180] or union of intrinsic characteristic of several images [99], [30], [187]. The source of images can be determined using these artifacts. In addition to methods mentioned above, research is primarily focused on methods based on artifacts generated due to manufacturing imperfections and in-homogeneity of camera sensors. In particular, Photo Response Non Uniformity (PRNU) has been widely popular for camera identification [3], [7], [34], [93], [69], [77], [114], [126], [186]. PRNU is stable with respect to change in environmental conditions. PRNU is unique and can be used to identify each device.

Estimation of PRNU signal from an image was first proposed by Lukáš *et al.* in [126]. According to [126], PRNU can be directly obtained by using the average of residual images. The residual image can be calculated by subtracting a denoised image obtained using wavelet denoising filter from an original image. Let $\mathbf{K}$ be PRNU and $\mathbf{W}_i$ be the residual extraction from i^{th} image of a given camera. Then, $\mathbf{K}$ can be obtained as

Figure 2.1: ***Original image (Left) and noise residue obtained after subtracting denoised image from original image. Image content is clearly seen in noise residue which can alter the correlation between reference PRNU and noise residue.***

$$\mathbf{K} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{W}_i \tag{2.1}$$

Fig 2.1 shows images under analysis and noise pattern obtained after subtracting denoised image from original image. The normalized correlation coefficient is then used to determine the source of a given image. However, the performance of correlation based detectors depends upon the robustness of extracted PRNU. The quality of estimated PRNU can be severely contaminated by image content, color interpolation, JPEG compression and other non-uniform artifacts. In addition, to get reliable and accurate results of identification, size of the estimated PRNU has to be large, for example, 512×512 pixels or above. However, due to the larger dimension of required PRNU, its use in image or video forgery localization results in poor performance [35]. Another example where high dimension of PRNU restricts its usage is camera fingerprints clustering [121], [24], [86]. Due to high dimension of PRNU, the complexity of clustering increases, which further leads to difficulties in computation and storage. The clustering algorithm is expected to use the lower dimension PRNU but still guarantee good performance. Therefore, exploring ways of improving the quality of PRNU extracted from small-sized image blocks is of great significance for the above mentioned PRNU-based applications.

There have been various attempts to improve the accuracy of source camera identification [16], [24], [30], [32], [44], [39], [84], [113], [35], [112]. These methods can be clustered into two groups.

In the first group, the algorithms aim to improve the estimate of PRNU. For example, Chen *et al.* [34] use the fact that PRNU is multiplicative noise, hence apply Maximum Likelihood Estimation (MLE) for estimation of a multiplicative factor from reference images. In [114], Li *et al.* show that for better reliability it is necessary to suppress unwanted artifacts from sensor

pattern noise. Hence, they propose to preprocess the PRNU by subtracting mean across rows and columns of PRNU (Zero Mean (ZM)) and applying Wiener filtering. Both processes remove the artifacts introduced by the camera processing pipeline. Hu *et al.* in [87] propose an enhanced method to extract camera sensor pattern noise (SPN). According to authors, reliable results can be obtained using correlation detection if the large component of a camera SPN is used while discarding the other components. They further expand their work from a single color channel (e.g., green channel) to three colors (RGB) channels in [85]. Li *et al.* in [114] propose that stronger components of PRNU are associated with scene details. The authors propose six enhancing models to reduce the interference from scene details. Color decoupled PRNU (CD-PRNU) extraction method is proposed in [113] by Li *et al.* Authors extract the PRNU noise patterns from each color channel and then fuse them to get a more reliable CD-PRNU. Kang *et al.* in [92] propose another enhancement method, where a camera reference phase PRNU is introduced to remove the periodic noise and non-white noise contamination in the reference PRNU. Effect of denoising filter is investigated in [42] by Cortiana *et al.*. The authors propose sparse 3d transform-domain collaborative filtering for more accurate noise extraction. According to Sutcu *et al.* [173], decision process can be enhanced by including camera's demosaicing characteristics. An effect of scenes and edges is suppressed by edge adaptive PRNU predictor proposed in [186] and [94]. The predictor is based on context adaptive interpolation (PCAI).

Since estimated PRNU can contain non-unique artifacts due to in-camera processing, several works have been proposed for pre-processing the estimated PRNU for removing non-unique artifacts. Lin *et al.* [119] propose to pre-process the estimated PRNU by identifying and suppressing the peaks according to local characteristics in the magnitude spectrum of the reference PRNU. They pre-process the reference PRNU with spectrum equalizer to remove the interfering periodic artifacts. Spectrum equalizer can be used with any PRNU estimation or enhancement methods, thereby improving the detection accuracy further.

In the second group, the methods aim towards improving the performance of SCI by using better statistical detection and similarity measures. Lukáš and Goljan [126] propose peak to correlation energy (PCE) measure. PCE attenuates hidden periodic patterns (a potential source of false identification), thereby increasing accuracy. In [92], Kang *et al.* use correlation over circular cross-correlation norm (CCN) to further decrease the false-positive rate.

An interesting application of PRNU is in indexing and searching images or videos from an extensive collection of data. The large PRNU dimensionality poses severe challenges and its compression for lower computational complexity is desired. Li *et al.* in [115], propose PCA-LDA based approach for compressing the PRNU to a large extent. They make use of PCA on noise residue for reducing the dimensionality by considering the eigen vectors with 99% variance. Further, they apply LDA by making use of labeled data

to compress the fingerprint further. Similar approach is followed at the time of testing.

Though all the references mentioned above use images of various formats for source camera identification, using raw data has not been deeply investigated except in [149]. The main focus of the problem is to coarsely identify the source device at model level. However, the algorithm may not be easily extended to obtain a finer device level identification. In another similar work, Thai *et al.* in [177], propose camera identification model based on hypothesis testing theory. They propose heteroscedastic noise model for raw images and estimate parameters for camera identification. However, the algorithm can differentiate between different devices of the same model but not different models or cameras.

2.0.2 Adaptive Contrast Enhancement-Based Counter Forensics

Image forgery is generally followed by techniques [153] that enhances the visual quality of the image and/or suppress the artifacts left by the forgery operation. Adversaries often use popular global contrast enhancement operations such as gamma correction (GC) and histogram stretching (HS) [48] for hiding the traces of manipulation done in an image. However, the global enhancement operations may not result in adequate perceptual quality. For example, in the case of HS, the quality of the image is hampered due to the quantum jump in cumulative distribution function (CDF) in the histogram after enhancement. In case of GC, the noise in the image may get amplified. On the other hand, Contrast-Limited adaptive histogram equalization (CLAHE) overcomes these limitations and prevents the over-amplification of noise. CLAHE is also suitable for improving the local contrast of an image. Usually, image forgery is created by modifying the image locally. Therefore, CLAHE is more useful in hiding the forgery in comparison to GC and HC.

Many techniques have been proposed to detect CE in images using first order statistics. For example, Stamm and Liu in [166] and [168] propose to detect CE in images by leveraging the peaks and gaps artifacts created by CE operation on the gray level histogram. In [169], the authors estimate whether an image is contrast enhanced and reconstruct the original image. In [25] and [28] also, peaks and gaps artefacts are exploited.

A lot of anti-forensics strategies have been proposed against such first-order statistics based CE detectors. In [26], Cao *et al.* propose local random dithering in the primary mapping of CE to avoid peaks and gaps in the histogram of the resulting enhanced image. Similarly, Kwok *et al.* in [108] use Internal Bit Depth Increase method to increase the precision in avoiding peaks and gaps artifacts. Also, Barni *et al.* in [11] propose a universal anti-forensics technique to counter histogram based manipulation detectors. Comesana-Alfaro and Perez-Gonzalez in [41] propose a general attacking method using

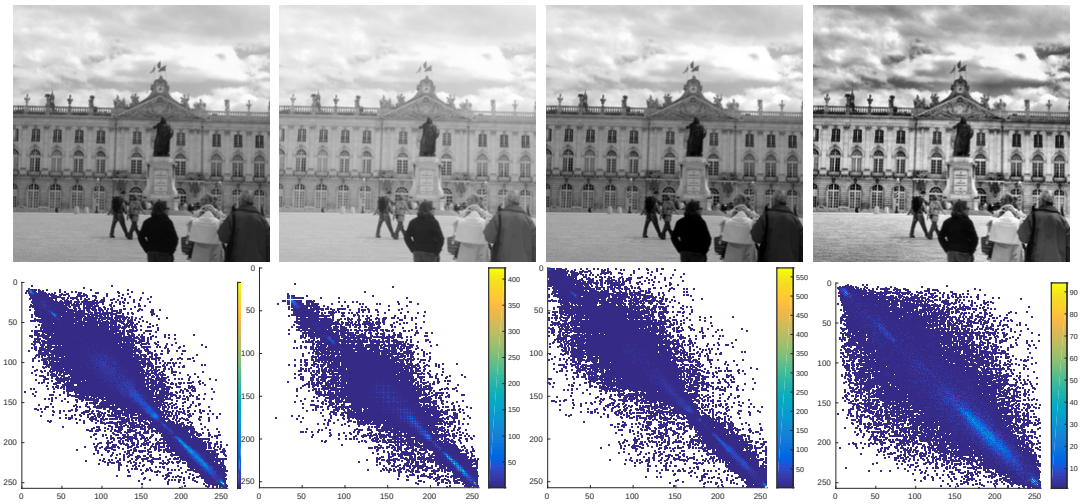


Figure 2.2: ***(Left to right) GLCM of Original image, enhanced images using gamma correction ($\gamma = 0.6$), histogram stretching and enhanced image using CLAHE (contrast enhanced limit 0.01) .***

a single target function against histogram based detectors. Cao *et al.* in [27] propose trace forging and trace hiding attacks against CE detectors.

Since first-order statistics based CE detection techniques are shown to be less robust by the above mentioned anti-forensics techniques [193], detectors based on other metrics are proposed. Authors in [120] use the inter-channel dependency because of color image interpolation to detect CE. Similarly, a recent algorithm proposed by De Rosa *et al.* in [193] looks at the Gray Level Co-occurrence Matrix (GLCM) of an image to determine whether it is enhanced or not. This uses second-order statistics of the image and is shown to be effective against anti-forensics techniques targeted at histogram based detectors. However, the accuracy of GLCM based detector reduces when images are enhanced using CLAHE since GLCM of images does not have gaps. Fig 2.2 shows GLCM of the original image, enhanced image using gamma correction ($\gamma = 0.6$), histogram stretching and enhanced image using CLAHE (contrast enhanced limit 0.01) respectively. It can be observed that gamma corrected, and histogram stretched images have gaps in GLCM whereas image created using CLAHE is free from gaps, and its GLCM looks similar to the original[1].

From the previous works, it can be observed that the detectors use artifacts in spatial domain [26, 108, 11, 41], [193]. However, steganalysis features proposed by Fridrich *et al.* such as Subtractive Pixel Adjacency Matrix (SPAM) [143], Spatial Rich Model (SRM) [72] and CC-PEV [144] can detect image manipulation operations such as contrast enhancement (like gamma correction or CLAHE) and compression with high accuracy in both spatial and DCT domains. SPAM and SRM features detect manipulations done in spatial domain whereas CC-PEV analyzes DCT domain for traces of tampering.

[1]For better visualization, please refer to soft copy version at 400% zoom.

Combining features from spatial and DCT domain [150], [9] gives more robust detector which can detect image manipulation in either domain.

The conventional CE forensics methods [48], [28] based on first or second order statistics, and steganalysis features [143], [72] utilize handcrafted features and have separate feature extraction and classification stages that cannot be simultaneously optimized. Therefore, to improve the performance, Sun *et al.* propose a Convolutional Neural Network (CNN) based detector which can also detect images that are manipulated by the counter forensics attack [171]. Similar approaches have been suggested in [160] and [189].

2.0.3 Egocentric analysis of dash-cam videos

The work presented in this section belongs to a class of problems called egocentric video analysis [65], [66], [67], [83]. Since this is a new problem, we only discuss the related works from other domains.

Peleg *et al.* [83] reveal one of the first applications in which the goal is to identify the film-maker from the biometric signature in ego-centric videos. Similarly, in [8], authors try to model artistic visual styles such as impressionism, baroque and cubism. According to the authors, every artwork has idiosyncratic signature which relates it to other works. They propose features derived from deep neural networks which allow the grouping of artworks into related art movements. Similar to the aforementioned idea, vehicle forensics serves to analyze egocentric biometric features [8], [31], [79], [90], [178] in videos or images. The aim of these works is to determine the identity of the person who captures the video/image or to identify artistic styles in paintings. On the other hand, vehicle forensics attempts to answer questions like, *"In which vehicle is the dash-cam mounted through which the video is recorded ?"*

Determining the painter of an artwork for forgery prevention and detecting fake artists has also been studied extensively. Several approaches for automatic style classification utilizing low level and object cues are presented in [31], [90], [178].

Determining who took a picture is a challenging and interesting problem. Towards this, the photographic style [178] and the location of the photograph [31], [79] can be used as cues for determining the identity of a photographer. The big disadvantage of these methods is that they are unable to distinguish between photographers using cameras on default settings (such as most wearable cameras) and at the same locations.

In the next chapter, we will briefly discuss the preliminaries required for better understanding of the problems.

3 Preliminaries

This chapter describes fundamental concepts for camera identification, antiforensics of contract enhancement and vehicle forensics. We also discuss approach used in state-of-the art algorithms for solving these problems. Since ego centric analysis of dash-cam video is a new work, we discuss closely related work which try to solve related problems in different domain.

3.1 Source Camera Identification

The existing approaches for camera identification can be grouped into three categories

- The first category examine Exchangeable Image File Format (EXIF) header or watermarks present in the images. Most camera attach EXIF header in their images. The information present in header consists of camera type, exposure, data and time about an image. However, this information can be maliciously altered and might not be available if the image is re-saved or re-compressed. The watermark embedded in images captured by some cameras can carry information about the biometric data of the photographer, camera, time and date. However, majority of camera manufacturers do not provide this option.
- In second category, difference in standard image processing methods such as demosaicing, gamma correction, color processing, white point correction and compression is considered. Image processing algorithms and processing steps vary from one manufacturer to another. Due to this difference, the output image may exhibit some traits and patterns regardless of the original image content.
- The third category uses the noise pattern of the imaging sensor of digital cameras. The noise pattern is generated due to pixel non-uniformity, dust on lens or dark currents. These noise patterns are unique for each camera and can be used as camera identification purpose.

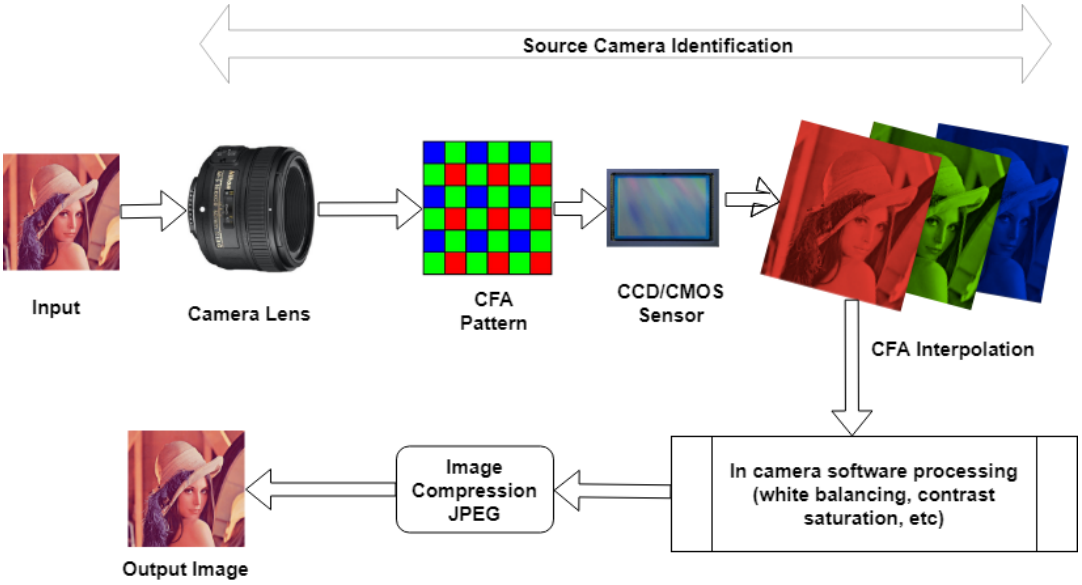


Figure 3.1: ***Color image pipeline. Typical components include lense, image sensor, Bayer filter, image scaling, gamma correction, image enhancement, color space conversion (between formats such as RGB, YUV or YCbCr), image compression (such as JPEG)***

3.2 Features generated in different stages of image acquisition

3.2.1 Image acquisition pipeline

In this section, we will briefly discuss the features embedded in the image at different stages of image generation. Fig 3.1 outlines the typical in-camera image processing pipeline. In every stage, image is embedded with artefacts which can be exploited for camera identification task. Prominent features used by forensic researchers are optical imperfections [91], [180], [99], [30], [187] color filter array/demosaicing [175] and senor pattern noise [3], [7], [34], [93], [69], [77], [114], [126], [186].

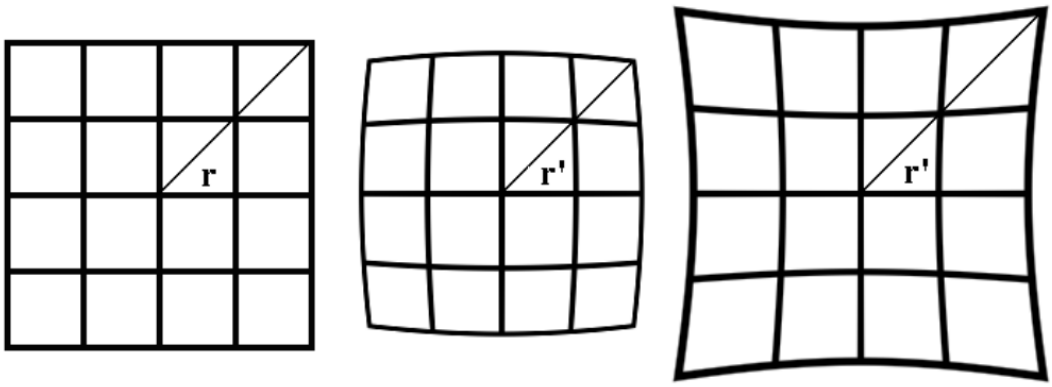


Figure 3.2: ***Radial distortion in the rectangular grid.***

Optical imperfections

In the first step of image acquisition, light rays pass through the optical lens (Fig 3.1). Due to the design and manufacturing process, lens produces aberrations in images [91], [180]. There are different kinds of lens aberrations such as spherical aberrations, coma, astigmatism, field curvature, radial distortion and chromatic distortion. The radial distortion is most severe among the different aberrations. The radial distortion causes straight lines in the object space rendered as curved lines on the film or camera sensor. Fig 3.2 shows the distortion of a rectangular grid. Camera manufacturers suppress this distortion by manipulating the system variables (indices, shapes, spacing, stops, etc). The degree and order of compensation vary from one manufacturer to the other or even in different camera models by the same manufacturer. As a result, lenses from different cameras leave unique imprints on the captured images. Radial distortion is computed from these imprints present in an image and can be used for identification of the camera model or device.

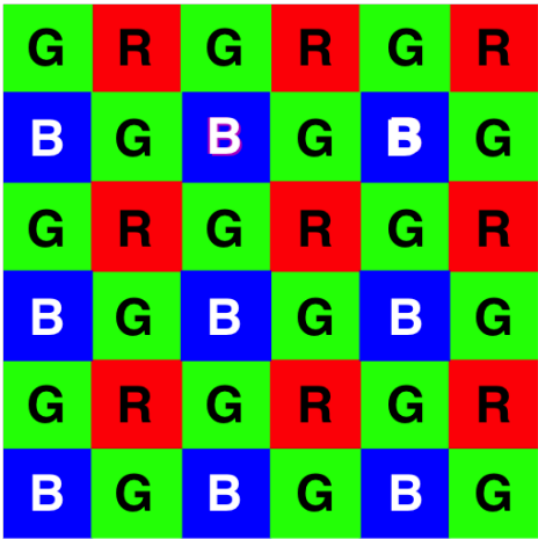


Figure 3.3: ***The Bayer color filter mosaic. Each two-by-two sub-mosaic contains 2 green, 1 blue, and 1 red filter, each filter covering one pixel sensor.***

Color Filter Array (CFA) and Demosaicing

Digital cameras capture imagery by covering the sensor surface with a color filter array (CFA) such that each sensor pixel only samples one of the three primary color values (Fig 3.3). To render a full-color image, an interpolation process, commonly referred to as CFA demosaicking [124], is required to estimate the other two missing color values at each pixel. Demosaicking consists of two successive steps: an interpolation for estimation of missing values by exploiting spatial and spectral correlation between the neighboring pixels and post processing step for suppressing the artifacts by means of median filtering. However, post processing is not always perfect and

resulting image can still have detectable demosaicing artifacts. Various features such as average pixel value, RGB pair correlation, center of mass of neighboring distribution, RGB pair energy ratio, wavelet domain statistics and a set of Image Quality metrics can use used along with the classifier for identifying the source of the camera of an image [98].

Senor Pattern Noise (SPN)

The captured digital image exhibit small change in intensity between individual pixels even if the sensor is capturing evenly lit scene. This change is due to presence of shot noise which is random in nature and partially due to the pattern noise which remains approximately same even if multiple images are taken of the same scene. The term "pattern noise" usually refers to two parameters. One is the DSNU (dark signal non-uniformity), which is the offset from the average across the imaging array at a particular setting (temperature, integration time) but no external illumination, and the PRNU (photo response non-uniformity), which describes the gain or ratio between optical power on a pixel versus the electrical signal output. The latter is often simplified as a single value measured at e.g. 50% saturation level, implying a linear approximation of the non linear PRNU. Since DSNU is additive in nature, some camera manufacturers suppress this noise by subtracting the dark frame from every image. The dominant part of SPN in natural images is PRNU. It is primarily due to different sensitivity of pixels to light caused by the inhomogenity of silicon wafers and imperfections during sensor manufacturing process.

In this thesis, we use PRNU for camera identification. We specifically work towards improving the estimation of PRNU. In next section, we discuss state-of-the-art algorithms for estimation of PRNU.

3.3 PRNU Estimation

In this section, we first briefly review the PRNU estimation procedure developed by Chen and Fridrich [34] and Kang [92] for source camera identification. We then discuss the applicability of these approaches to SCI.

3.3.1 Maximum Likelihood Estimation

Chen and Fridrich [34] propose the following linear model for sensor output of a camera:

$$\mathbf{I} = \mathbf{I}^0 + \mathbf{I}^0\mathbf{K} + \Theta \tag{3.1}$$

where $\mathbf{I}$ denotes camera output, $\mathbf{K}$ is PRNU factor, $\mathbf{I}^0$ is the scene image without any camera noise and Θ is the combination of independent random noise components such as dark current, shot noise and quantization noise.

Signal-to-noise ratio (SNR) between observed image $\mathbf{I}$ and PRNU factor $\mathbf{I}^0\mathbf{K}$ can be improved by suppressing $\mathbf{I}^0$ from $\mathbf{I}^0\mathbf{K}$ by using host signal rejection. Towards this, denoised version of image $\mathbf{I}$ is subtracted from both sides in Eq. (3.1), $\hat{\mathbf{I}}^0 = F(\mathbf{I})$, is obtained using denoising filter F.

$$\mathbf{W} = \mathbf{I} - \hat{\mathbf{I}}^0 = \mathbf{I}\mathbf{K} + \mathbf{I}^0 - \hat{\mathbf{I}}^0 + (\mathbf{I}^0 - \mathbf{I})\mathbf{K} + \Theta = \mathbf{I}\mathbf{K} + \theta \tag{3.2}$$

In Eq. (3.2), term θ is the sum of Θ and two other components introduced by denoising filter. Image content is significantly suppressed in noise residue $\mathbf{W}$. Hence, PRNU is estimated using $\mathbf{W}$ instead of $\mathbf{I}$.

With assumption that camera is available, estimator of PRNU factor is obtained from N images $\mathbf{I}_1$, $\mathbf{I}_2$,...., $\mathbf{I}_N$. Noise term θ can be modeled as White Gaussian Noise (WGN) with variance σ^2. Energy of PRNU $\mathbf{I}\mathbf{K}$ signal is small as compared to noise term, hence it is reasonable to assume that noise component θ is independent of $\mathbf{I}\mathbf{K}$. From Eq. (3.2), for each $k = 1, 2, ..., N$,

$$\frac{\mathbf{W}_k}{\mathbf{I}_k} = \mathbf{K} + \frac{\theta_k}{\mathbf{I}_k} \tag{3.3}$$

The log-likelihood of observing $\frac{\mathbf{W}_k}{\mathbf{I}_k}$ given $\mathbf{K}$ is,

$$L(\mathbf{K}) = -\frac{N}{2}\sum_{k=1}^{N} log(\frac{2\pi\sigma^2}{\mathbf{I}_k^2}) - \sum_{k=1}^{N}\frac{(\frac{\mathbf{W}_k}{\mathbf{I}_k} - \mathbf{K})^2}{\frac{2\sigma^2}{\mathbf{I}_k^2}} \tag{3.4}$$

The estimate $\hat{\mathbf{K}}$ is obtained using MLE and is given by:

$$\hat{\mathbf{K}} = \frac{\sum_{k=1}^{N}\mathbf{W}_k\mathbf{I}_k}{\sum_{k=1}^{N}\mathbf{I}_k^2} \tag{3.5}$$

We will refer PRNU estimated using Eq. (3.5) as MLE-PRNU [34].

Kang *et al.* in [92] propose that SPN can be contaminated largely in the frequency domain by image content and non-unique artifacts of JPEG compression, on-sensor signal transfer, sensor design and color interpolation. Since PRNU can be modeled as white noise signal [126], it is reasonable to assume that the frequency spectrum will be flat. To remove the contamination due to image details and camera signal processing in frequency domain, authors in [92] propose the following procedure for estimation of PRNU.

The noise residue W_k is whitened first in the frequency domain and has constant Fourier magnitude coefficients except that its DC Fourier coefficient equals zero. Phase only component $W_{\phi k}$ is extracted as follows:

$$\begin{aligned} W_k &= DFT(W_k) \\ W_{\phi k} &= W_k/|W_k| \end{aligned} \tag{3.6}$$

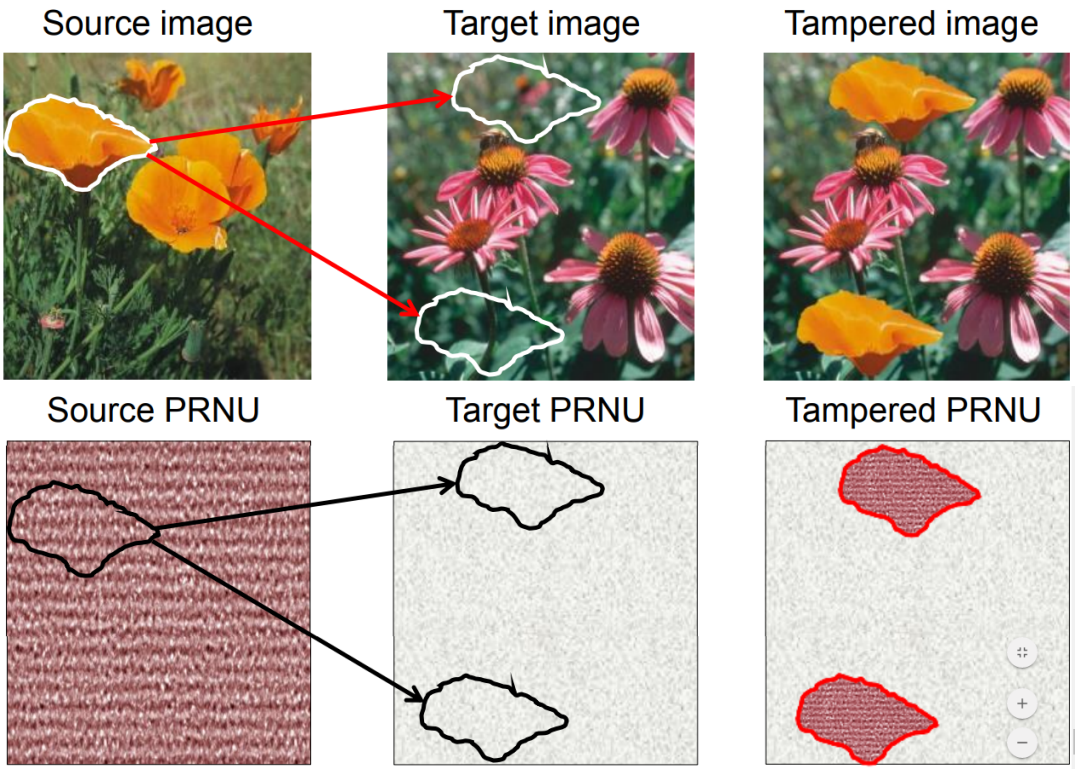


Figure 3.4: ***The PRNU pattern is unique and spatially related to each region in an image. It is present in all pristine images produced by the camera but absent in tampered areas. By detecting the presence/absence of the camera PRNU in the image under test, a decision of forgery can be obtained.***

where $|W_k|$ is the Fourier magnitude of W_k.

The PRNU $\mathbf{K}$ is obtained by averaging the phase-only component $|W_{\phi k}|$ and performing inverse DFT.

$$\mathbf{K} = real\left(IDFT\left(\frac{\sum_{k=0}^{N-1} W_{\phi k}}{N}\right)\right) \tag{3.7}$$

where N is the total number of images used in extraction process. From here on, we will refer PRNU estimated in Eq. (3.7) as Phase-PRNU

Analysis of PRNU for forensic application is not only limited to source camera identification. PRNU signatures are also used for localizing forgeries in images. However reliable tamper localization is still a challenge, especially for small size forgeries. A conventional approach for localization of forgery is to compare the detector's response with a threshold obtained after modeling the decision statistics. Fig 4.3 illustrates scheme for detecting copy move forgery using PRNU. Since PRNU pattern is spatially related to the region in an image, it can be observed that by detecting the presence or absence of the PRNU in the image we can detect the copy move forgeries.

3.4 Contrast Enhancement Counter Forensics

Existing studies [26, 108, 11, 41, 27] on contrast enhancement counter forensics either focus on removing peaks and gaps in the histogram (first-order statistics) of contrast-enhanced images or removing the gaps in GLCM of images (second-order statistics). In this section, we will discuss approaches used by previous works in this field.

In [11], Barni *et al.* propose a universal counter forensic technique based on first-order statistics for concealing the traces left by enhancement operations such as contrast enhancement. According to authors, the adversary will process the image and then perform slight modifications on the resulting images such that the histogram of the manipulated image is as close as possible to that of the original image while maintaining high fidelity between processed and original image. In this attack, adversary first performs histogram retrieval by searching for most similar histogram from a database of untouched images. In the second step, the histogram of an attacked image is modified such that it is close to histogram retrieved in the first step while satisfying the constraints on the maximum distortion due to histogram mapping. Finally, pixels in the attacked image is changed according to the histogram mapping keeping distortion as low as possible. However, the assumption of easy accessibility to a database of untouched images with the similar histogram to that of an image to be attacked may be restraining. Fig 3.5 shows the difference in the histogram of an attacked image before (CE image histogram) and after (Remapped image histogram) the antiforensic operation.

In [27], authors generate contrast-enhanced images which are undetectable by forensic detectors by integrating local random dithering into the pixel values. Traditional CE operation is decomposed into two steps. In the first step, the mapping function $f(.)$ transforms an integer pixel value x into a real number and in the second step, obtained pixel value is rounded to yield the resulting pixel value y. Mathematically, enhancement operation is given by

$$y = round(f(x)) \tag{3.8}$$

It can be observed in Fig 3.6-(b) that histogram peak/gap bin would appear when multiple/none gray levels x are mapped into the unit neighboring range of y [27]. To avoid generating empty or accumulated bins, authors propose to add local random dithering into the primary mapping. That is

$$y = round(f(x) + n) \tag{3.9}$$

where the random variable n has Gaussian distribution with zero mean and σ standard deviation. The resulting antiforensic images do not contain gaps as demonstrated in Fig 3.6-(c) and (d). One of the significant advantages of

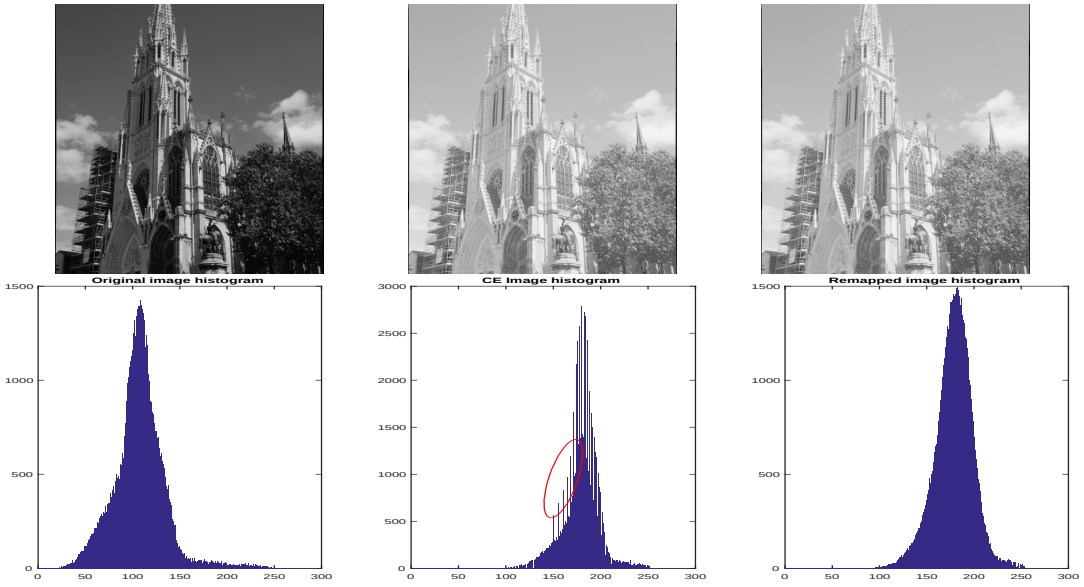


Figure 3.5: ***(Left to right) Original image, enhanced image using gamma correction ($\gamma = 0.6$), and antiforensically enhanced image generated using [22] along with the respective histograms. Peaks are visible in the histogram of gamma corrected image (peaks are marked in red), whereas, the histogram of the remapped image is free of peaks.***

[11] and [27] is that the histogram of the attacked image looks quite similar to that of the original image. Thus, the first order detectors based on histogram cannot easily distinguish between attacked and original images.

In a similar work proposed in [151], Hareesh *et al.* formulate counter forensics of contrast enhancement as an optimization problem using a variant of the Total Variation (TV) norm. By solving optimization equation given by Eq 3.10, authors generate enhanced image Y that looks like contrast enhanced version of the original image X. In addition, Y would be similar to X in terms of natural image statistics.

$$\min_{Y} \frac{w_1}{2}\|Y - X\|_2^2 + \frac{w_2}{2}\big\|Y - \phi(X)\big\|_2^2 + (|\nabla_z Y| + |\nabla_y Y|) \tag{3.10}$$

where w_1 and w_2 are regularization parameters and $\phi(.)$ is contrast enhancement (CE) function which is known beforehand. The first term in Eq 3.10 is for getting an enhanced image Y closer to X, the second term ensures Y is closer to the contrast-enhanced image, and third term is TV norm regularization term. Fig 3.6 shows image generated using Eq 3.10 and its histogram. Antiforensically generated image has histogram similar to that of original image and visually looks similar to the conventionally enhanced image. Since the gray level co-occurrence matrix (GLCM) of attacked image is similar to that of original image, the method performs well against GLCM based detectors.

The motivation behind the proposed approach is to develop an effective antiforensic algorithm for adaptive contrast enhancement. The aforementioned

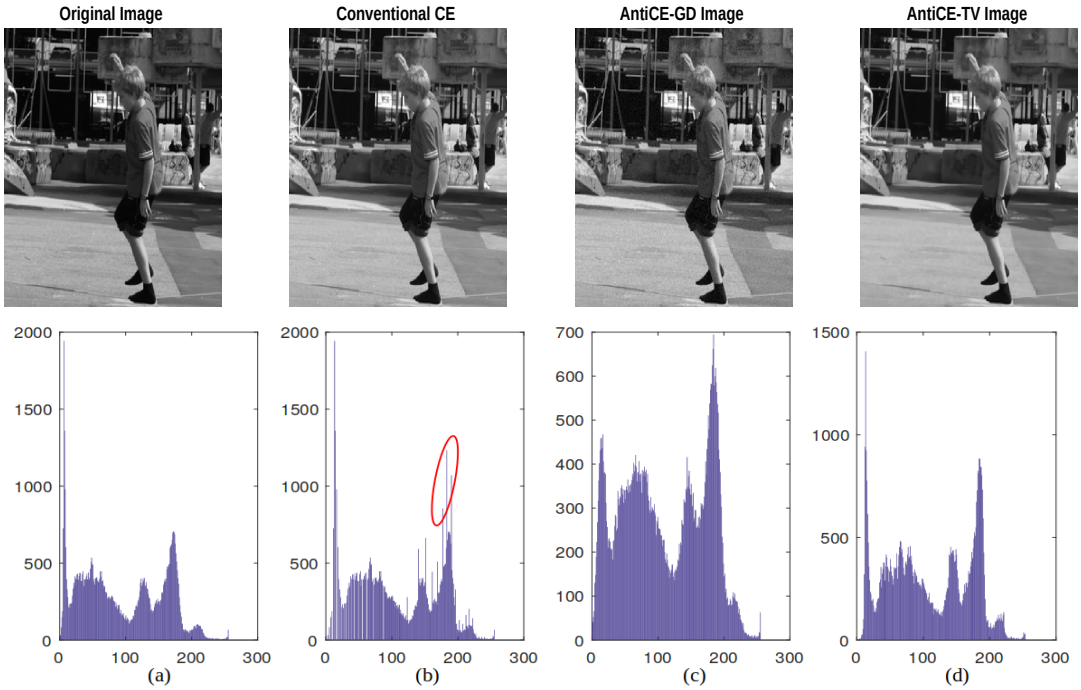


Figure 3.6: ***(Left to right) Original image, enhanced image using gamma correction ($\gamma = 0.6$), and antiforensically enhanced [27] (AntiCE(GD)), [151] (AntiCE(TV)) along with the respective histogram in the bottom row. Histogram of gamma corrected image shows peaks and gaps (peaks are marked in red), whereas, histograms of antiforensically generated images are free of peaks.***

anti-forensic methods mainly focus on hiding the artifacts in the spatial domain. However, there are detectors, such as CC-PEV, which can detect the enhancement in the frequency domain also. Further, the algorithm proposed in [11] requires knowledge of image histogram, whereas, [27] adds random noise to images. By adding dithering noise to the image, the algorithm compromises on visual quality of the image. In case of [151], the TV norm term in Eq 3.10 smoothens the image too much. To overcome these limitations, the proposed algorithm exploits knowledge of both spatial and DCT domains simultaneously for generating the enhanced images. The algorithm further employs Huber Markov Random Field for preserving the natural statistics of the image.

3.5 Egocentric analysis of dash-cam videos

For vehicles, the motion makes the camera jittery, and the video recorded by the camera will be degraded by motion blur. Motion blur, which is caused by the dynamics of the vehicle, is unique in nature and can be used to identify the vehicle accurately. This unique signature is different for each vehicle due to the difference in mechanical and the structural properties [137]. Identification of a vehicle using vibrometry data has been a well-studied research problem [95], [163], [162]. These algorithms require Laser

Radar (LADAR) to record the vibrometry data. Mechanical vibrations due to the engine, the mechanical structure and motion of a vehicle produce an opportunity to detect and identify vehicles with different engine types and configurations as they embed different vibration characteristics. A vehicle's surface vibrations can be remotely recorded using a continuous wave fixed-frequency LADAR [95], [163]. Commercial application of these methods is limited since LADAR is not installed in every car and the use of such sensors is not popular. Hence, it is important to use sensors which are popular among the users of the vehicles. In our case, it is the video recorded from dash-cams. There is no prior work related to the identification of vehicles using egocentric videos (video captured using dash-cams).

On similar lines, gait recognition [131] is done on video from wearable cameras to determine a person's body shape and dynamic walking style. Body mounted accelerometers have also been used for recognition of the person wearing it [138].

Dash-cam videos are also used in the field of transportation [191], [181]. In [191], Yuan *et al.* proposed a video-based road detection algorithm via online structural learning. They make use of local gradient and texture features such as DSIFT (DenseSIFT), HOG (Histogram of Oriented Gradient) and LBP (Local Binary Pattern) for transforming the road detection problem into detecting the road boundary. Similarly in [181], Wang *et al.* extract deep features for better representation of high-level information for congestion detection. Traditional and deep features discussed in above-mentioned works can detect roads, vehicles, and congestion with high accuracy. However, these algorithms extract semantic features and represent an image scene. Therefore, their relationship with the vibratory and motion signature of the vehicle cannot be directly established. For example, if we extract LBP from a video frame, it will provide us with the texture information rather than the motion of the vehicle. While keeping in mind the limitation of these approaches for the problem in hand, we use motion blur as the primary feature as it inherently captures the vibration and motion of the vehicle.

The core of such a system is the need for reliable separation of the signal of interest i.e., blur due to motion and vibration of the vehicle from other sources of blur such as uneven roads, light conditions, different pressure of air in tires and due to motion of other objects in the view. The task of subtle signature detection is complicated by the variability of other factors and due to the small amplitude of the signal of interest that is buried in the noise or hidden in the complex blur due to various other factors. For understanding the complexity of the problem, let us consider the blurred frame F given by:

$$F = k_1 * k_2 * I + \theta \tag{3.11}$$

Where k_1 is a signal of interest i.e., blur due to subtle motion and vibration of the vehicle and k_2 is a blur due to the rest of the factors previously discussed.

Estimation of k_1 is an ill-posed because we only observe blurred frame F and all other variables (k_1, k_2, I) are unknown.

The challenges faced in the extraction of k_1 in equation 3.11 is similar to the reliable extraction of the fetal signal from the maternal contribution [2, 96, 82]. In fetal ECG detection problems, a small amplitude fetal complex is often buried in the noise or embedded inside the large-amplitude maternal complex. The algorithms for extracting fetal contributions are based on enhancing the fetal or maternal contribution by passing the combined signal from the low pass filter. Similarly, for extracting features related to k_1, we can extract low-frequency components from time-series data of joint blur kernels and use them as discriminative features. The change in a blur due to other factors such as uneven roads will result in higher jitters. Such type of jitters will introduce higher than a normal variation in the blur, which is absent when the road is smooth. Hence, the higher frequency can be considered as noise. Whereas blur due to subtle motion and vibration of the vehicle will be captured by low frequency due to their continuous presence at all duration's. We will further discuss the feature extraction algorithm in Chapter 6.

The subsequent chapters details the proposed methods for problems discussed in this chapter and outline the comprehensive experiments and results.

4 Source Camera Identification and Tamper Detection

This chapter outlines the proposed PRNU estimation algorithm. In Section 4.1.1, we first discuss the conversion of JPEG to raw, followed by PRNU extraction process in Section 4.1.2 . Raw captures image data recorded by sensor. Since the raw data does not undergo any in-camera image processing, we can obtain robust estimate of PRNU using raw compared to final JPEG output images. Then, in Section 4.1.5, we explain the identification procedure which uses the correlation between noise residue from a given test image and camera reference pattern. We further discuss the elaborate experimental results in Section 4.2.2. Forgery detection and localization are discussed in Section 4.3. In the same section, we also present results for some realistic forgeries.

4.1 Proposed Algorithm

4.1.1 JPEG to raw conversion

To obtain probabilistic estimate of RAW measurements from available JPEG images from camera c, we use algorithm proposed by Ayan *et al.* in [32]. According to authors, inverse RAW distributions from calibrated data can be estimated using following steps, (a) calibrated data is used for obtaining forward deterministic tone-map for a given camera and (b) the model is inverted probabilistically. Authors in [32] use similar approach for modeling camera processing pipeline proposed by Kim *et al.* in [102]. Deterministic forward map $\mathbb{J}: g \rightarrow x$ from raw tricolor sensor measurements is obtained for a pixel $g \in [0,1]^3$ using its corresponding rendered JPEG color values $x \in \{[0,255] \bigcap \mathbb{Z}\}^3$. After obtaining forward map, model is inverted probabilistically to obtain the inverse probabilistic distribution $p(g|x)$.

$$p(g|x) = \frac{1}{\Phi} p(g) \exp\left(-\frac{||x - J(g)||^2}{2\sigma_f^2}\right) \tag{4.1}$$

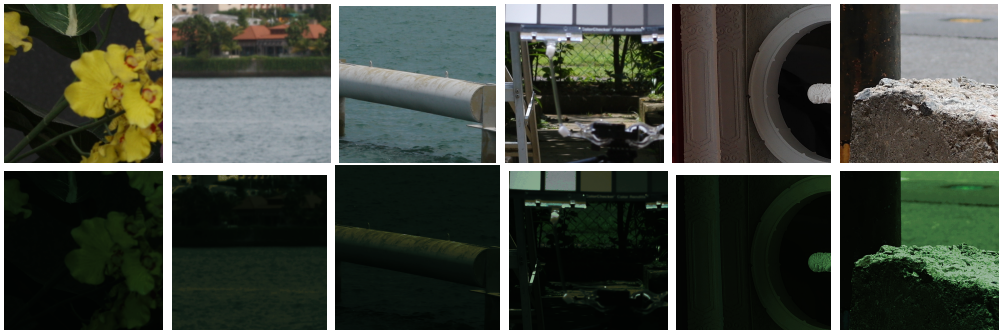


Figure 4.1: ***JPEG images (first row) with corresponding estimated RAW images (second row).***

where Φ is normalization factor

$$\Phi = \int p(g^{'}) \exp\left(-\frac{||x - J(g^{'})||^2}{2\sigma_f^2}\right) dg^{'} \tag{4.2}$$

$p(g)$ is prior on sensor measurements. Conditional distribution $p(g|x)$ given by Eq. (4.1) also accounts for errors in the estimate J of the rendering function, treating them as Gaussian noise with variance σ_f^2. A uniform prior is chosen over all possible sensor measurements whose chromaticities lie in the convex hull of training data. With the knowledge of conditional distribution $p(g|x)$, the raw data corresponding to given JPEG value can be estimated.

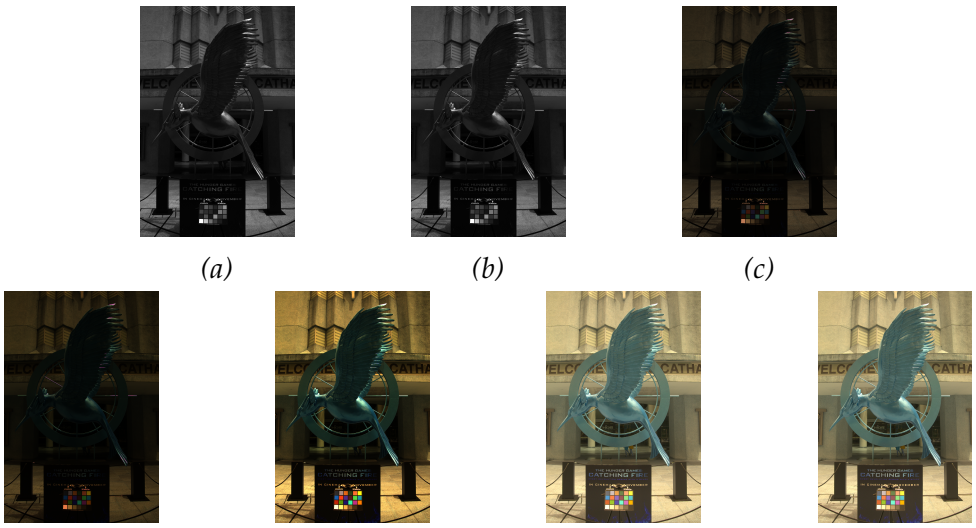


Figure 4.2: ***Output image at each step of camera's image processing pipeline (Samsung NX2000). (a) Raw image with value between dark level and saturation level (b) linearized raw data $m \times n$ array with real values between 0 and 1 (c) $m \times n \times 3$ Demosaiced image (d) sRGB color space image, (e) Brightened sRGB image (f) Gamma corrected image (g) JPEG compressed image with Quality factor of 90.***

For creating training data which is used for determining the probabilistic inverse of a particular camera, a large number of RAW-JPEG pairs are required. Our database consists of images captured by cameras listed in Table 4.1 using XRite-24 patch color checker. Most of the cameras in our data set allow access to RAW sensor measurements, and therefore directly give us a set of RAW-JPEG pairs for training. We have over 5000 RAW-JPEG pairs per camera for estimating probabilistic inverse. Fig. (4.1) shows JPEG images (first row) and their corresponding probabilistically obtained RAW images (second row). The RAW images usually appear darker compared to their JPEG image counterpart. Raw sensor data is a $m \times n$ array with integer values between black and saturation level of the camera sensor. These values are different for different camera models. For example, Canon 600D has black and saturation level values between 2047 and 13584 whereas for Samsung NX2000 these values are between 0 and 4095 respectively. Fig (4.2) shows the output of each image processing step for Samsung NX2000. Since these values are not very bright compared to the full uint16 range of 0-65335, the raw images are mostly dark (Fig. (4.2a) and (4.2b)). Whereas JPEG image (Fig. (4.2g)) undergoes additional processing such as gamma correction, white balancing, image sharpening, noise reduction, color space conversion and demosaicing. DCRAW is used for processing the image using parameters available in EXIF data of the image[1].

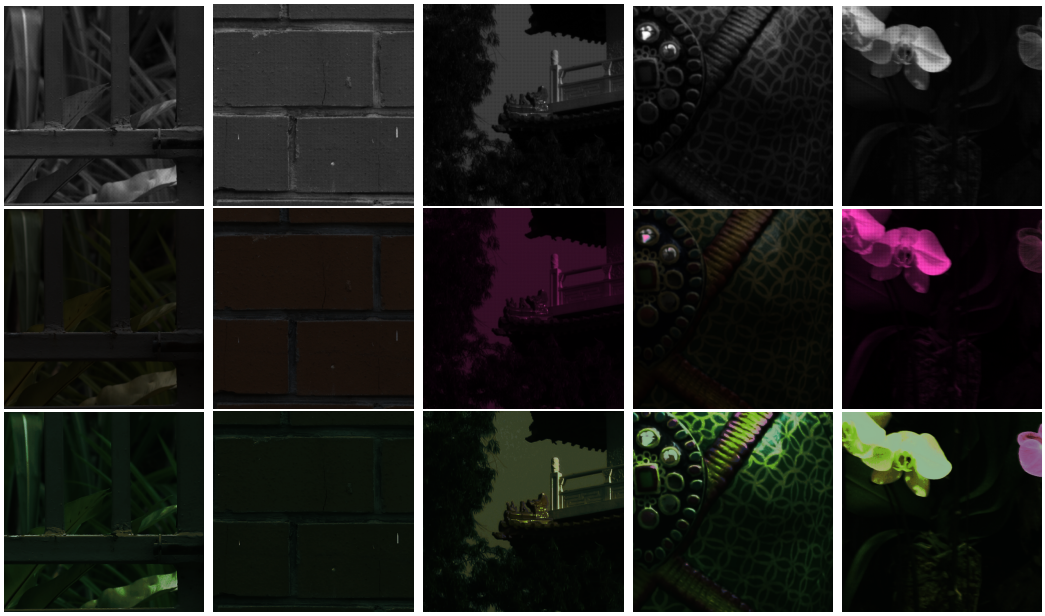


Figure 4.3: ***First row: TIFF images obtained using dcraw [170], Second row: raw images after processing TIFF images and Third row: estimated raw images using [32]***

For determining the quality of estimated raw images, we compute mean square error (MSE) between images obtained using DCRAW and estimated

[1]The exact replication of camera pipeline is not possible since manufacturers do not disclose all the algorithms they have used for generating the image.

raw images. DCRAW converts raw image into TIFF/PPM format. TIFF image is further processed (Linearizing and Demosaicing) [170] to obtain ground truth raw image. In Fig. (4.3) first row shows TIFF images obtained after processing corresponding raw images[2]. Second row of Fig. (4.3) is obtained by processing the tiff images shown in first row, whereas probabilistically estimated raw images using [32] are shown in third row. MSE between images of second and third row is calculated. For images shown in Fig. (4.3) from left to right MSE is 0.0017, 0.011, 0.0128, 0.0041 and 0.0177 respectively. Also, the average mean square error for randomly selected 500 images is 0.009, which is low. Hence we can infer that estimated raw images are close to actual raw images. Same fact can also be validated by referring to [32], where authors show the effectiveness of estimated raw values for different vision applications. Since exact camera pipeline cannot be simulated using software, there will be a visual difference between estimated and actual raw images [170]. We can observe the difference in 3rd and 5th images of second and third rows in Fig. (4.3).

4.1.2 PRNU estimation from raw data

Let $I \in \mathbb{Z}^+$ be an image of size $m \times n$ and $\mathbf{I}$ be its vectorized version of size $M \times 1$ where $M = mn$. $\mathbf{I}$ is converted to raw data $\mathbf{G}$ using Eq. (4.1). Let $\mathbf{G} = \{g_i\}\ \forall\ i = \{1, 2, ..., M\}$, where g_i denotes raw value at i^{th} pixel, $\mathbf{K} = \{k_i\}\ \forall\ i = \{1, 2, ..., M\}$, k_i is PRNU factor at i^{th} pixel and $\mathbf{F} = \{f_i\}$, $\forall$ $i = \{1, 2, ..., M\}$, f_i is true value at i^{th} pixel. Sensor output at i^{th} pixel can be modeled as.

$$g_i = k_i f_i + \eta_i \tag{4.3}$$

where, η_i is Poisson noise.

For estimation of PRNU, we assume that we have multiple images taken from the camera. If there are N images available from a particular camera, then we have N observations for each pixel. Since image acquisition is a Poisson process, we can define the joint probability of the i^{th} pixel from N images as:

$$p(\mathbf{g}|\mathbf{kf}) = \prod_{j=1}^{N} \frac{(k_{ij} f_{ij})^{g_{ij}} \exp(-k_{ij} f_{ij})}{g_{ij}!} \tag{4.4}$$

subscript ij denotes i^{th} pixel in j^{th} image. Subscript j can be dropped from k_{ij} since PRNU at i^{th} location is same for all images. Now, PRNU can be

[2]TIFF images have been obtained from the original proprietary raw format using DCRAW utility (with parameters -D -4 -T -j -v. Refer to Appendix A for more details.)

computed using MLE. Taking log likelihood of Eq. (4.4),

$$\begin{aligned}\log(p(\mathbf{g}|\mathbf{kf})) &= \sum_{j=1}^{N} \log\left(\frac{(k_i f_{ij})^{g_{ij}} \exp(-k_i f_{ij})}{g_{ij}!}\right) \\ &= \sum_{j=1}^{N} g_{ij} \log k_i f_{ij} - \sum_{j=1}^{N} k_i f_{ij} - \sum_{j=1}^{N} \log g_{ij}!\end{aligned} \tag{4.5}$$

Differentiating Eq. (4.5) w.r.t k_i

$$\sum_{j=1}^{N} \frac{g_{ij} f_{ij}}{k_i f_{ij}} - \sum_{j=1}^{N} f_{ij} = 0 \tag{4.6}$$

Rearranging terms in 4.6, k_i is given by

$$k_i = \frac{\sum_{j=1}^{N} g_{ij}}{\sum_{j=1}^{N} f_{ij}} \tag{4.7}$$

4.1.3 Estimating true sensor data from noisy sensor data

It can be observed from Eq. (4.7) that true pixel value is required. Since true pixel value is not available, we can compute an estimate of true value. The true value is corrupted by noise generated in the sensor during image acquisition process. Eq. (4.3) can be expanded as follows

$$g_i = (1 + l_i) f_i + \eta_i \tag{4.8}$$

$$g_i = f_i + \Theta \tag{4.9}$$

where l_i is component of PRNU and Θ is $l_i f_i + \eta_i$.

De-noising of raw data is done using CBM3D algorithm [45]. For denoising, CBM3D function makes use of noisy RGB channels of raw data along with standard deviation of AWGN. Noise content in an image depends on the type of camera used for capturing an image. It is observed that variance of noise (σ_n^2) is less in DSLR cameras. This is because DSLR cameras may incorporate efficient algorithms to attenuate noise to have high-quality images. We test our algorithm for different values of variance. It is observed that cameras capable of storing both RAW and JPEG perform better with σ_n equal to 1, while cameras capable of storing only JPEG images gives better performance with σ_n equal to 10. Θ is assumed to be AWGN and can be removed from $\mathbf{G}$ using CBM3D to obtain an estimate of f. The resulting

true value estimate $\hat{f}$ is substituted in Eq. (4.7) in place of f to obtain k_i.

$$k_i = \frac{\sum_{j=1}^{N} g_{ij}}{\sum_{j=1}^{N} \hat{f}_{ij}} \tag{4.10}$$

Eq. (4.10), is used to calculate PRNU for three channels (Red, Green and Blue). Finally, PRNU $\mathbf{K}$ is obtained using following rule.

$$\mathbf{K} = 0.3 * \mathbf{K}_{red} + 0.6 * \mathbf{K}_{green} + 0.1 * \mathbf{K}_{blue} \tag{4.11}$$

Where $\mathbf{K}_{red}$, $\mathbf{K}_{green}$, $\mathbf{K}_{blue}$ are the estimates of PRNU obtained using red, green and blue channel respectively.

4.1.4 Cramer Rao lower bound

Cramer Rao lower bound (CRLB) on variance of k_i can be obtained by calculating second derivative of Eq. (4.5). CRLB on variance is:

$$var(k_i) \geq \frac{1}{-\mathbb{E}\left(\frac{\partial^2 \log(p(\mathbf{g}|\mathbf{kf}))}{\partial^2 k_i}\right)} \tag{4.12}$$

$$\frac{\partial^2 \log(p(\mathbf{g}|\mathbf{kf}))}{\partial^2 k_i} = -\frac{\sum_{i=1}^{N} g_{ij}}{k_i^2} \tag{4.13}$$

Hence CRLB bound on variance is given by

$$var(k_i) \geq \frac{\mathbb{E}\left(k_i^2\right)}{\sum_{i=1}^{N} g_{ij}} \tag{4.14}$$

since k_i is a zero-mean noise-like signal [34] with some variance σ^2, Eq. (4.15) reduces to.

$$var(k_i) \geq \frac{\sigma^2}{\sum_{i=1}^{N} g_{ij}} \tag{4.15}$$

which is consistent with the bound proposed in [34]. Also since our model is linear, proposed Maximum likelihood (ML) estimator is minimum variance unbiased (MVU) and its variance is determined by CRLB, $var(k_i) \sim \frac{1}{\sum_{i=1}^{N} g_{ij}}$. Hence, we can infer that a large number of images used for estimation results in tighter bound.

4.1.5 Decision Statistics

In order to determine if image under investigation is taken from a specific camera, decision statistics between noise residue and camera reference pattern is computed. Noise residue is calculated as $\mathbf{W} = \mathbf{G} - \hat{\mathbf{F}}$, where $\hat{\mathbf{F}}$ is estimate of true values. The normalized correlation coefficient can be used as the test statistic [126], [34], [92]. The normalized correlation coefficient $\rho(\mathbf{W}, \mathbf{K})$ between the image noise residue $\mathbf{W}$ and the camera reference PRNU $\mathbf{K}$ is defined as

$$\rho(\mathbf{W}, \mathbf{K}) = \frac{\mathbf{WK}}{\|\mathbf{W}\|\|\mathbf{K}\|} \tag{4.16}$$

where $\mathbf{WK} = \sum_{i=0}^{N-1} W_i K_i$, $\|\mathbf{W}\|$ is the L2-norm of $\mathbf{W}$.

Goljan *et al.* [126], propose another decision statistics, peak correlation energy (PCE) to measure the similarity between $\mathbf{W}$ and $\mathbf{K}$. The circular shift vector is $y_m = \{y_{i \oplus m}\}$ where the operation $\oplus$ is modulo N addition in $\mathbb{Z}_N$. The circular cross-correlation $r_{\mathbf{NK}}(m)$ is defined as:

$$r_{\mathbf{WK}}(m) = \frac{1}{N}\mathbf{WK}_m = \frac{1}{N}\sum_{i=0}^{N-1} W_i K_{i \oplus m} \tag{4.17}$$

PCE is defined as

$$PCE = \frac{r_{\mathbf{WK}}^2(0)}{\frac{1}{N-|A|}\sum_{m \notin A} r_{\mathbf{WK}}^2(m)} \tag{4.18}$$

where A is a small neighbor area around zero where $r_{\mathbf{WK}}^2(0) = \frac{1}{N}\sum_{i=0}^{N-1} W_i K_i$, $|A|$ is the size of A.

The correlation over circular cross-correlation norm (CCN) can be used to further decrease the false-positive rate:

$$CCN = \frac{r_{\mathbf{WK}}^2(0)}{\sqrt{\frac{1}{N-|A|}\sum_{m \notin A} r_{\mathbf{WK}}^2(m)}} \tag{4.19}$$

Since the noise residual also contains the PRNU component, correlation $\rho(\mathbf{N}, \mathbf{K})$, peak correlation energy $PCE(\mathbf{W}, \mathbf{K})$ and circular cross correlation norm $CCN(\mathbf{W}, \mathbf{K})$ will be high when the given image is captured from the corresponding camera from which reference PRNU is computed. For our experiments we use PCE as decision statistics, however similar results can be obtained using other decision statistics.

4.1.6 Hypothesis Testing Formulation

The decision statistics obtained in Section 4.1.5 is used in formulating a hypothesis for testing the source camera of the image. To reliably decide whether the given image is captured from a particular camera, we need to determine the distribution of decision statistics under different hypothesis and corresponding thresholds. Towards this, let Δ denote decision statistics coefficient when image $\mathbf{I}$ is taken from the same camera and $\bar{\Delta}$ is decision statistics coefficient if an image is not taken from the same camera. We will determine the distribution of Δ and $\bar{\Delta}$ theoretically. The threshold is obtained by solving binary hypothesis problem and performance is evaluated correspondingly.

Noise residue for each patch is extracted by subtracting denoised patch from original patch and correlation between noise residue and camera reference pattern of each patch for all cameras is computed. Let $\mathbf{\Delta} = \Delta(i)$, $\forall\ i \in \{1, ..., t\}$, denotes vector of decision statistics coefficients between noise residue of all patches and camera reference pattern belonging to same camera, where t is number of such patches. Also, $\bar{\mathbf{\Delta}} = \bar{\Delta}(i)$, $\forall\ i \in \{1, ..., r\}$, is vector of decision statistics coefficients between noise residue of all patches and camera reference pattern belonging to different camera and r is number of such patches. In this case, r will be $c - 1 \times t$ where c is the number of cameras needed to match. For deciding whether a patch belongs to a particular camera, we solve the following binary hypothesis problem. H_0 describes null hypothesis and H_1 represents alternative hypothesis.

$$\begin{aligned} H_0 &:= \textit{Image is taken by camera } c \\ H_1 &:= \textit{Image is not taken by camera } c \end{aligned} \tag{4.20}$$

Probability density function (pdf) under both hypothesis are calculated by modeling the distribution of $\mathbf{\Delta}$ and $\bar{\mathbf{\Delta}}$ as Generalized Gaussian, given as

$$f(\Delta; \alpha, \beta, \mu) = \frac{\beta}{2\beta\Gamma(\frac{1}{\beta})} \exp\left(-\frac{|\Delta - \mu|^{\beta}}{\alpha}\right) \tag{4.21}$$

where $\Gamma(\bullet)$ is the Gamma function given by $\Gamma(x) = \int_0^{\infty} \exp^{-t} t^{x-1} dt\ (x > 0)$ μ, α, $\beta > 0$ denotes mean, scale and shape parameter respectively and α is given by $\alpha = \sigma_{GGD} \sqrt{\Gamma(1 \setminus \beta) \setminus \Gamma(3 \setminus \beta)}$ (σ_{GGD} is standard deviation of GGD). The parameters are estimated using MLE [142][3].

[3]Detailed derivation of parameters are given in Appendix C

Table 4.1: ***Details of Cameras Used for Experiments***

Camera Brand	Sensor	Native Resolution	Image Format Supported
Canon1DsMkIII	36 x 24 mm CMOS	5616 × 3744	CR2(RAW)+JPEG
Canon 60oD	22.3 x 14.9 mm CMOS	5184 × 3456	CR2(RAW)+JPEG
Canon EOS 40D	22.2 x 14.8 mm CMOS	3888 × 2592	CR2(RAW)+JPEG
Canon PowerShot S90	1/1.7" CCD	3648 × 2736	CR2(RAW)+JPEG
Fujifilm J10	1/2.5" CCD	3264 × 2448	JPEG
Fujifilm XM1	23.6 x 15.6mm CMOS	4896 × 3264	RAW+JPEG
Nikon D5200	23.5 x 15.6 mm CMOS	6000 × 4000	RAW + JPEG
Olympus EPL6	17.30 x 13.00 mm CMOS	4608 × 3456	RAW +JPEG
Panasonic DMC LX3	1/1.63" CCD	3648 × 2736	RAW+JPEG
Panasonic GX1	17.3 x 13 mm CMOS	4592 × 3448	RAW+JPEG
Samsung NX2000	23.5 x 15.7 mm CMOS	5472 × 3648	RAW+JPEG
Sony A57	23.7 x 15.6 mm CMOS	4912 × 3264	RAW+JPEG

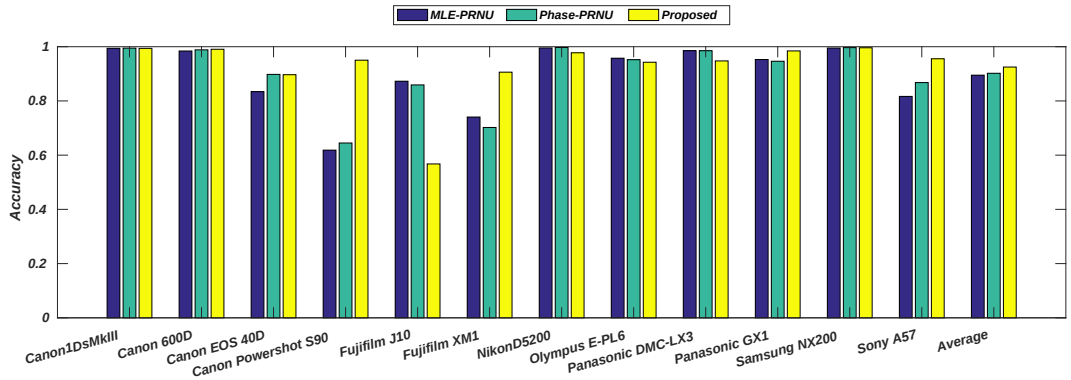
4.2 Experiments Setup and Results

4.2.1 Experimental Setup

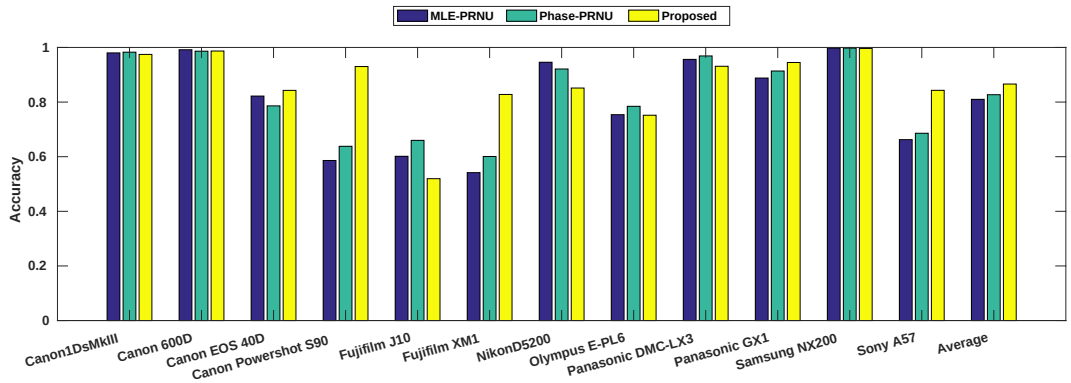
Cameras of different brands and model with different sensor size, resolution and image format are used for experiments. Table 4.1 lists details of cameras used[4]. From each camera, 50 images are randomly selected for estimation of PRNU and 150 images for camera identification. Experiments are performed on patch size of various dimensions. Selected images from the listed cameras are divided into patches of size varying from 64×64 to 512×512 and reference PRNU for each patch location is estimated. From each camera, 50 images which are selected for estimation of PRNU are first converted to raw data and then proposed algorithm, as discussed in Section 4.1 is used for estimation of PRNU.

Once we obtain PRNU, we evaluate our estimation method with algorithms discussed in Section 3.3 of Chapter 3. We apply both Zero-Mean (ZM) and Wiener filter (WF) operation for MLE-PRNU [34] and Phase-PRNU [92] to improve the performance of source identification. However, we do not apply WF operation for proposed method. WF operation is applied for removing artifacts present in estimated PRNU [34]. Since, we estimate the PRNU from probabilistically obtained raw data, our estimate is free from artifacts introduced by in-camera processing. Also for comparing the obtained estimate with other popular methods [34], [92], we do not apply ZM and WF for our experiments. We directly use obtained estimate as reference PRNU for camera identification without any preprocessing. All the experiments are performed under the same conditions.

[4]Images can be downloaded from http://www.comp.nus.edu.sg/~whitebal/illuminant/illuminant.html



(a) 512 × 512



(b) 256 × 256

Figure 4.4: ***Classification accuracy per camera for patch size*** 512 × 512 ***and*** 256 × 256

4.2.2 Classification Results

In this section, we compare results of our proposed algorithm with PRNU estimation algorithms, i.e., MLE-SPN and Phase SPN. Using hypothesis formulation as explained in Section 4.1.6, the decision threshold is calculated for each camera for fixed false positive rate (FPR). FPR is kept below or equal to 1%. The threshold is used to determine detection accuracy which is calculated using following equation.

Table 4.2

Actual (True) Condition / **Evaluation Conclusion**	Image is taken by camera c	Image is not taken by camera c
Image is taken by camera c	True Positive	False Positive
Image is not taken by camera c	False Negative	True Negative

$$Acc(\%) = \frac{tp + tn}{tp + tn + fp + fn} \tag{4.22}$$

where tp, tn, fp and fn denotes the statistics of the observed true positive, true negative, false positive and false negative respectively.

Table 4.3: ***Classification accuracy per camera for patch size*** 128×128

Camera Name	SPN Estimation Method		
	MLE-PRNU [34]	Phase-PRNU [92]	Proposed
Canon1DsMkIII	0.9095	0.8961	0.8566
Canon 60oD	0.9665	0.9623	0.9427
Canon EOS 40D	0.7129	0.7063	0.7698
Canon Powershot S90	0.6044	0.6409	0.9016
Fujifilm J10	0.5275	0.517	0.4959
Fujifilm XM1	0.5161	0.5368	0.7042
NikonD5200	0.7	0.7188	0.654
Olympus E-PL6	0.5702	0.5714	0.61
Panasonic DMC-LX3	0.9299	0.925	0.9046
Panasonic GX1	0.7032	0.7322	0.8177
Samsung NX200	0.9883	0.9901	0.9916
Sony A57	0.5257	0.5448	0.5916
Average Accuracy	**0.721**	**0.728**	**0.770**

We present the results in Fig. (4.4a)-(4.4b) and Tables 4.3-4.4, we note that our method performs better than MLE-PRNU and Phase-PRNU. From Fig. (4.4a) the mean detection accuracy for proposed algorithm is 92.5% whereas for MLE-PRNU and Phase-PRNU accuracy is 89.5% and 90.2% respectively. Similar results can be observed from Fig. (4.4b) and Table 4.3 for patch size 256×256 and 128×128 respectively. Proposed algorithm also performs better for small size patches. For example, from Table 4.4 we can observe that average accuracy of the proposed algorithm is 66.6% in comparison with 64.44% and 64.1% for MLE-PRNU and Phase-PRNU respectively, which is a significant improvement since the number of 64×64 patches used for testing is large (around 0.25 million). Similar results can be inferred for patch size of 128×128 (Table 4.3), where improvement in detection accuracy is over 5%.

Since Fujifilm J10 is JPEG-only camera, raw proxy[5] is used to obtain probabilistic inverse. According to [32] the quality of probabilistic inverse required for converting JPEG to RAW is substantially lower when using a RAW proxy. Therefore there is more uncertainty in raw values obtained for JPEG-only camera in comparison with RAW-capable camera. Thus, we observe sub-par results for Fujifilm J10.

Images are also compressed for efficient storage. The compression introduces quantization noise which degrades the PRNU estimation. To study

[5]Interested readers can refer to [32] for more details about how to create raw proxy.

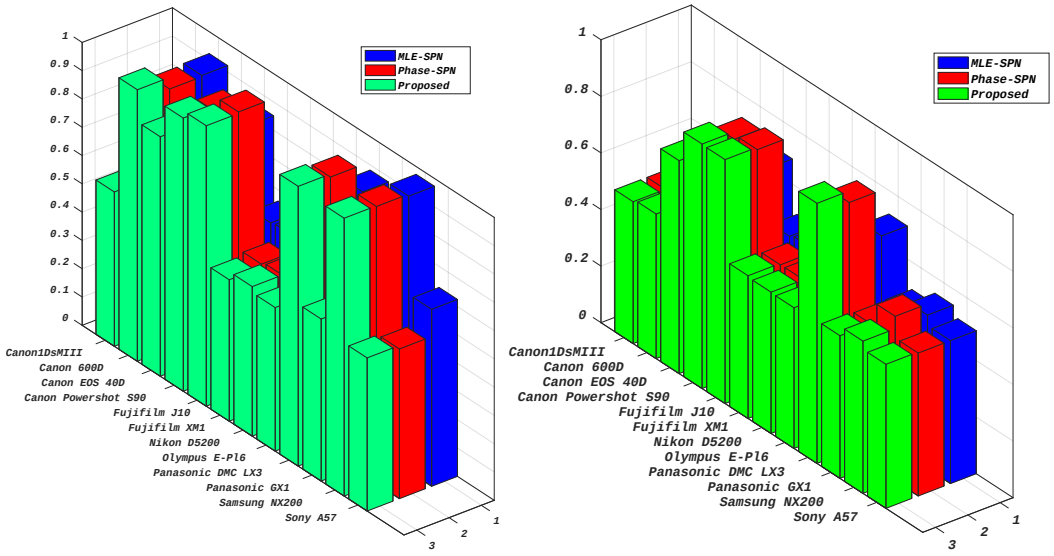


Figure 4.5: ***Classification accuracy per camera for patch size*** 512×512 ***(Left) and*** 256×256 ***(Right) with quality factor of*** 40 ***and*** 10 ***respectively.***

the effect of compression on proposed algorithm, test images from each camera are compressed using JPEG compression with quality factor of 40 and 10. Fig. (4.5) show the accuracy of different PRNU estimation techniques for patch size of 512×512. When quality factor is 40, proposed algorithm has average accuracy of 74.40%, whereas MLE-SPN and Phase-SPN have average accuracy of 68.44% and 72.98% respectively. Similarly for quality factor of 10, proposed, MLE-SPN and Phase-SPH have average accuracy of 62.22%, 61.18%, 54.73% respectively. This drop in accuracy is expected because of interference of quantization noise with PRNU component.

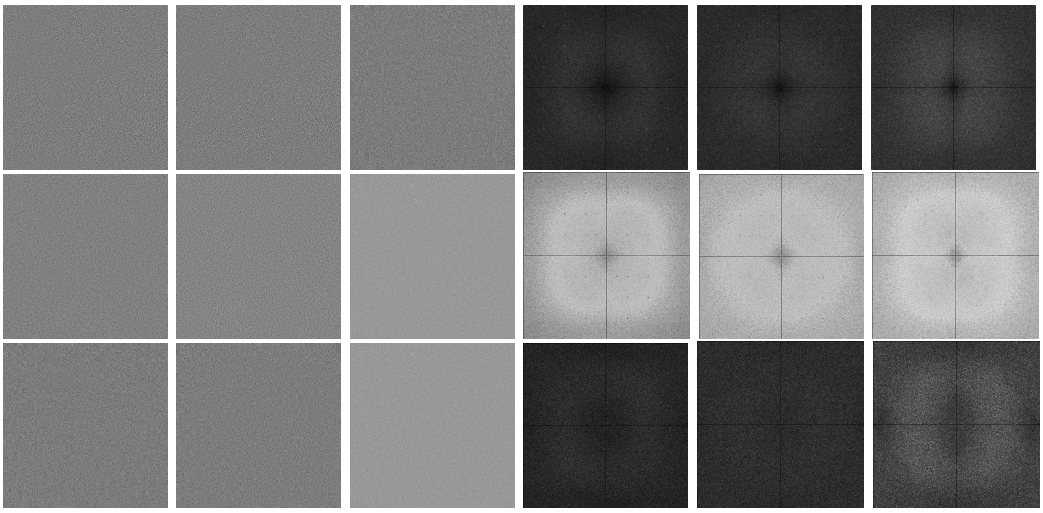


Figure 4.6: ***Spatial domain (first three columns) and frequency domain (last three columns) representation of PRNU estimated by proposed (first row), MLE-PRNU (second row) and Phase-PRNU (third row) respectively for 3 different cameras***

PRNU is considered as high-frequency noise. Previous work on estimation

Table 4.4: ***Classification accuracy per camera for patch size*** 64×64

Camera Name	SPN Estimation Method		
	MLE-PRNU [34]	Phase-PRNU [92]	Proposed
Canon1DsMkIII	0.7247	0.7278	0.6579
Canon 60oD	0.8415	0.8396	0.7975
Canon EOS 40D	0.6289	0.6079	0.6312
Canon Powershot S90	0.6064	0.6384	0.8003
Fujifilm J10	0.509	0.5195	0.5001
Fujifilm XM1	0.5102	0.5072	0.6163
NikonD5200	0.5301	0.5342	0.5365
Olympus E-PL6	0.5094	0.5152	0.524
Panasonic DMC-LX3	0.8279	0.8406	0.8301
Panasonic GX1	0.5558	0.5458	0.6
Samsung NX200	0.9456	0.9419	0.9448
Sony A57	0.5106	0.5161	0.5296
Average Accuracy	**0.641**	**0.644**	**0.664**

of PRNU shows that estimate obtained using MLE-SPN and Phase-SPN is contaminated by image content which is also high frequency. Therefore for analyzing the frequency content of PRNU, we plot the estimated PRNU obtained using different methods for three cameras. Fig. (4.6) shows the PRNU for three cameras in spatial (first three columns) and frequency (last three columns) domain[6]. First row shows estimate obtained using the proposed algorithm. If we compare the plots for high-frequency content, we can observe that very high frequency which is likely to be image content is suppressed in the proposed algorithm. However, estimate obtained using MLE-PRNU is contaminated by high frequency. Phase-PRNU is an improvement over MLE-PRNU, but very high frequency is still present. Therefore, from these plots, we can infer that estimate of PRNU obtained using proposed algorithm consists of less image content in comparison to other two methods.

In order to further improve the accuracy, various enhancements are proposed as discussed in introduction section of this thesis. In one such enhancement, Sutcu *et al.* [173] propose to use PRNU based camera identification method along with camera identification method based on CFA interpolation artifacts [16]. According to authors, verifying the consistency of demosaicing artifacts can further lead to improvement in identification

[6]For better visualization, please refer to soft copy version.

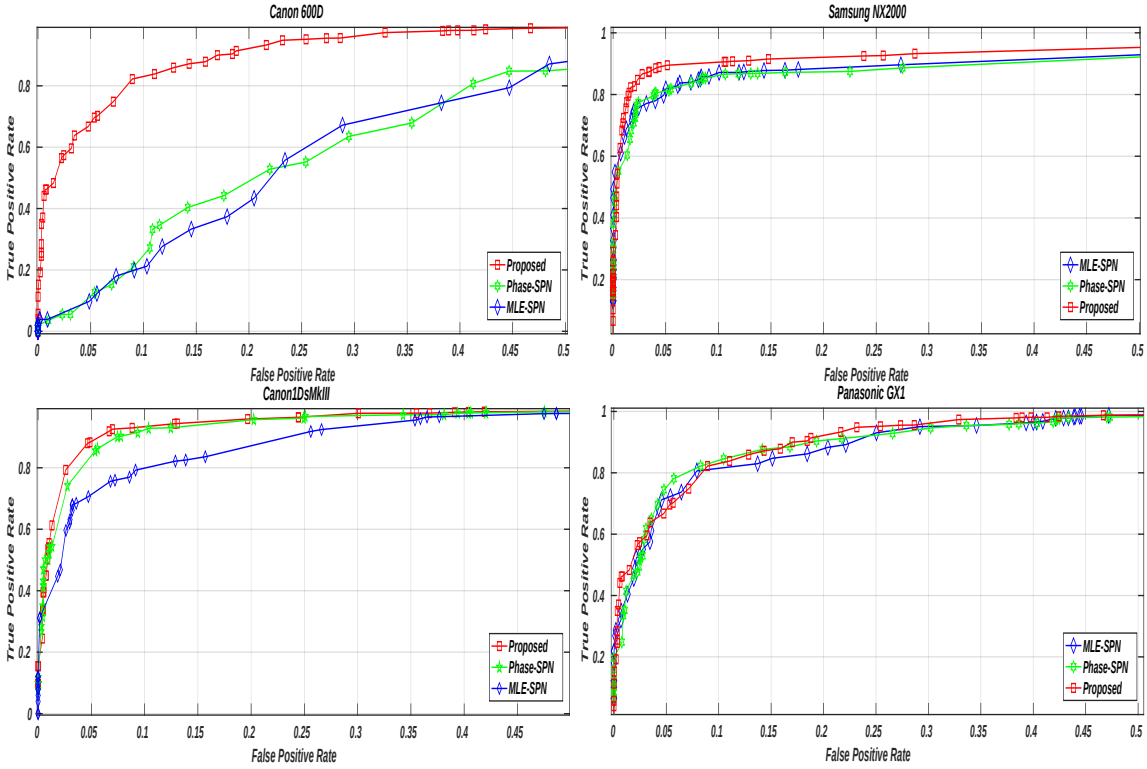


Figure 4.7: ***ROC plots for detection of synthetic forgeries with the*** 128×128 ***size window.***

accuracy. In [173], authors use two-step verification for identifying the acquisition device of the image. In the first step, camera identification using PRNU estimation technique proposed in [20] is performed. If PRNU does not match with noise pattern of the image, it is decided that image is not taken by the camera under investigation. However, if the first step is successful, then in second step, images are further tested using classifier for verifying the consistency of demosaicing artifacts. Demosaicing artifacts are extracted using algorithm proposed in [146]. The final decision is made by classifier. We perform similar experiments for Canon EOS 40D and Sony A57, however we use PRNU estimated using proposed algorithm in first step. We observe that for patch size of 128×128, identification accuracy for Canon EOS 40D increased from 76.98% to 77.89% for fixed threshold (PCE) of 9 whereas, for Sony A57 there is an increase of 2.04% from 59.16% to 61.20% for threshold (PCE) of 4. It should be noted that threshold is selected such that false positives before applying two step procedure is around 1%. This marginal improvement in accuracy is due to further decrease in false positives. However, better estimation of PRNU is desirable to achieve good results in first step. Hence proposed algorithm can also be used in conjunction with various enhancement algorithms for improving the accuracy further.

4.3 Forgery Detection

The forgery detection at each pixel is formulated as hypothesis (eq. 4.20) testing applied to a block surrounding the pixel. Current algorithms test for the presence of PRNU in each sliding window separately and finally fuse all local decisions. Local decisions are made based on obtained correlation between residual image extracted by the sliding window and corresponding PRNU. To obtain reliable statistics, correlation should be computed over a sufficiently large window. However, to detect small size forgery, it is required to keep window size as small as possible. Hence there is a trade off between window size and size of forgery detected. If we want to detect small forgery, correlation over a small window should be good enough to give reliable results.

Camera Name	#Images used for Estimation of PRNU	#Images used for Estimation of Predictor
Canon1DsMkIII	50	20
Canon 600D	50	20
Panasonic GX1	50	20
Samsung NX2000	50	20

Table 4.5: ***Model of cameras used for testing forgery localization and number of images used per camera for estimation of PRNU and Predictor.***

Ability to detect small size forgeries depends on analysis window [34], [39], [40], [84], [104], [129]. However, in this thesis, we want to analyze the improvement in localization performance due to improved PRNU estimate only. Better PRNU estimate results in more reliable correlation which in turn leads to better tamper localization.

Several algorithms focus on improving the decision by improving PRNU estimate. This can be achieved by suppressing image content by using more reliable denoising filters [39], [45]; equalizing the spectrum of the PRNU estimate [119]; retaining only its phase content information (Phase-PRNU) [92], [114]; attenuating strong components from image content [115]; or suppressing color interpolation artifacts based on color filter array's structure [85]. However, it should be noted that above-mentioned methods can be considered as post-processing for improving the performance after estimating the PRNU. Since we evaluate the robustness of our proposed estimation method, we do not use any post-processing for comparing the results. Results of localization are compared with PRNU estimated using MLE-SPN [34] and Phase-SPN [92]. We use segmentation-guided PRNU analysis proposed in [104] for localizing forgeries in synthetic and realistic images.

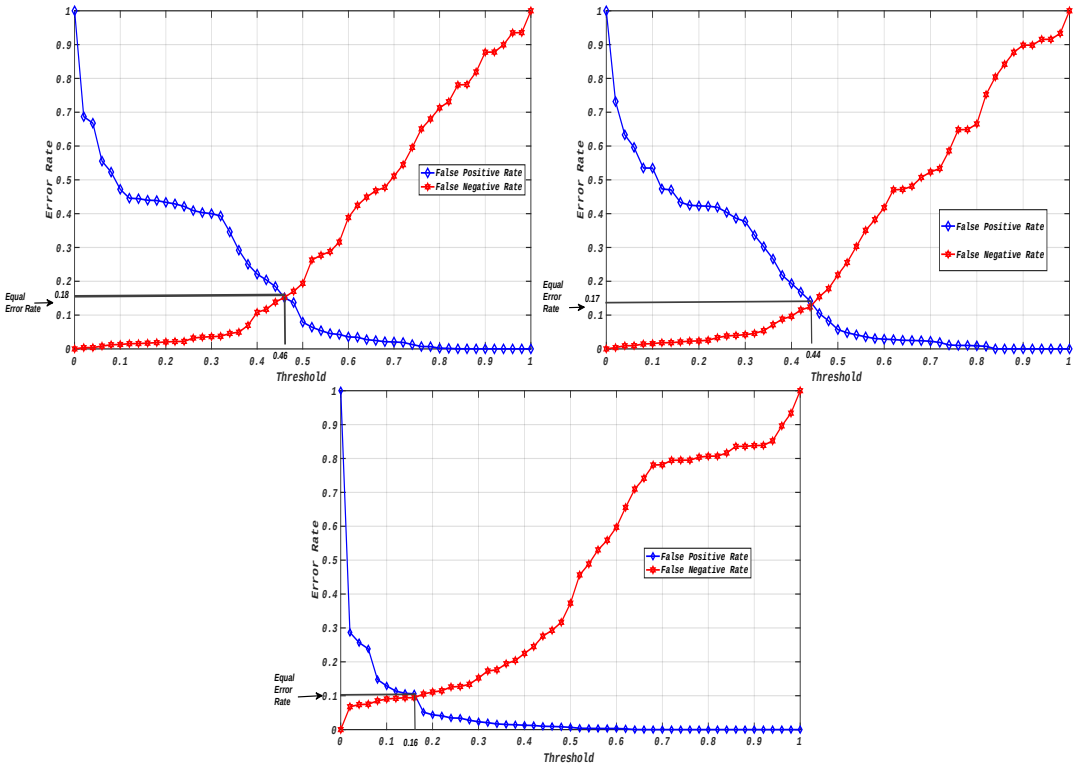


Figure 4.8: ***Error rate plots for Cannon 60oD. MLE-SPN, Phase-SPN and Proposed algorithm (left to right) respectively.***

Experimental Results

Although the size of an analysis window plays important role in detecting forgeries, for simplicity we use analysis window of size 128×128. The choice of window size is inspired by [34], which established that 128×128 is large enough for calculating reliable statistics and also able to detect small forgeries. However, windows of another size can also be used. Detailed analysis related to window size can be found in [104].

For each camera listed in Table 4.1, we randomly select 50 images and create synthetic forgeries of different size. From each image, we crop rectangular patch of size 64×64 or 128×128 from a random location and replace it with same size patch from images of other cameras. Three different correlation predictors [34] are created using PRNU estimated by proposed, MLE-PRNU, Phase-PRNU respectively. The output of segmentation guided PRNU analysis is a probability map. A decision map is further obtained using probability map by applying a suitable threshold. We uniformly select 50 threshold values between 0 and 1 and plot ROC for each estimation method. Fig. (4.7) shows the ROC from detection of synthetic forgeries for four cameras. We can observe from Fig. (4.7) that correlation predictor created using PRNU estimated from proposed algorithm outperforms predictors created using MLE-PRNU and Phase-PRNU methods. We observe that the performance of Canon 60oD has significantly improved (Fig. 4.7). The size of the sensor pixels in this camera (refer Table 4.1 for pixel size details) is smaller

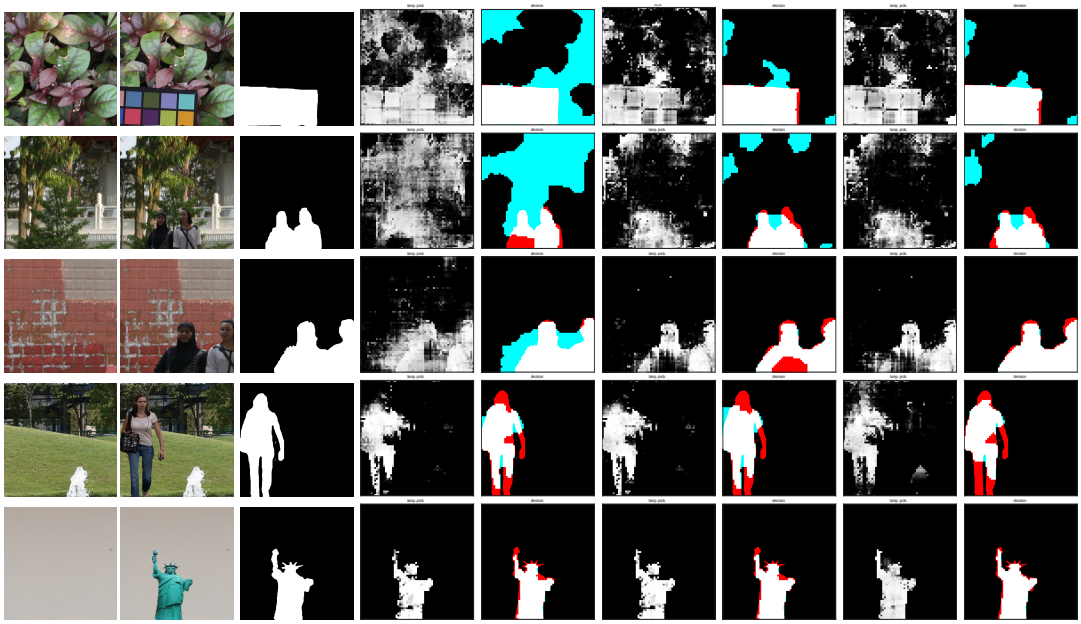


Figure 4.9: ***Localizing forgeries on realistic images using MLE-PRNU*** *(*4^{th} ***and*** 5^{th} ***column), Phase-PRNU*** *(*6^{th} ***and*** 7^{th} ***column), proposed*** *(*8^{th} ***and*** 9^{th} ***column).*** 1^{st} ***column shows original images,*** 2^{nd} ***column tampered images,*** 3^{rd} ***column ground truth respectively. Whereas green color indicates false positives and red true negative.***

than other cameras leading to better results. Estimate of PRNU depends on the size of pixels also, if the size of pixel is large, PRNU will be averaged and after in-camera operations, PRNU signal is further degraded. Since MLE-PRNU and Phase-PRNU estimate the PRNU from JPEG or TIFF images, their estimate of PRNU for such cameras may not be good. However, the proposed algorithm estimates PRNU from raw data and therefore gives better estimation of PRNU. In order to further analyze the performance of proposed algorithm we plot error rate curve for Cannon 60oD. Fig. (4.8) shows Equal Error Rate (EER) and the corresponding threshold for MLE-SPN, Phase-SPN and proposed algorithm. From Fig. (4.8) we can observe that EER for proposed algorithm is 0.1 which is lower as compared to other algorithms. We have used the machine with 2.2 GHz 4th generation core i7 processor for performing the experiments. The average time taken by the algorithm for detecting the forgery that is not larger than 128×128 is 8.52 seconds per image. For all cameras, reference PRNU is estimated apriori and stored in the hard disk.

Fig. (4.9) shows performance of proposed algorithm for realistic forgeries. Decision maps for images ($5^{th}, 7^{th}$ and 9^{th} column) is constructed using threshold of 0.5 over probability map ($4^{th}, 6^{th}$and 8^{th} column). We can observe that decision maps obtained when proposed algorithm (8^{th} and 9^{th} column) is used as estimation algorithm for detecting forgeries is closer to ground truth ($3^{rd}column$) than decision map obtained using MLE-PRNU (4^{th} and 5^{th} column) or Phase-PRNU (6^{th} and 7^{th} column). There is a considerable improvement in tamper localization which is evident from lower value

of false positives (green region in decision map) and higher value of true positives for an image. For example, in Fig. (4.9) for 2^{nd} image (second row), fp in case of [34], [92] and proposed algorithm is 0.5175, 0.1079 and 0.0685 respectively. Similarly for 6^{th} image (sixth row), tp for MLE-PRNU, Phase-PRNU and proposed algorithm is 0.8494, 0.8533 and 0.8842 respectively with zero fp. Similar results are observed for other images too.

This chapter covered novel method for estimation of PRNU. The main focus of the chapter was on camera identification, tamper detection and localization. In the next chapter, we will discuss the counter forensic approach for adaptive contrast enhancement.

5 Adaptive Contrast Enhancement Counter Forensics

This chapter outlines joint spatial and DCT optimization framework for generating antiforensically enhanced images. Having argued in Chapter 1 that counter forensic approach is useful in improving the reliability of forensic methods, in this chapter, we evaluate our algorithm using various contrast enhancement forensic detectors. We also report results of comparison with other popular counter forensics algorithms discussed in Section 3.4 of Chapter 3. For successful Contrast enhancement Counter Forensics image prior plays vital role. In Section 5.1, we discuss briefly choice of prior for proposed algorithm. Sections 5.2 and 5.3 outline proposed algorithm and experimental results followed by discussion in Section 5.4.

5.1 Natural Image Prior

Forensic detectors for identifying enhanced images mostly rely on the statistical properties of the image. These detectors exploit the fact that when an enhancement is done, it will lead to a change in the statistical properties of an image. For generating antiforensically modified images that are undetectable to forensic detectors, their statistical properties should be similar to natural images. Therefore, a good image prior is of utmost importance.

Good priors have been applied to different tasks such as image denoising [147, 23, 50, 81, 46, 156], image inpainting [156] and other applications [106], yielding excellent results. However, learning good priors from natural images is a daunting task - the high dimensionality of images makes learning, inference and optimization with such priors prohibitively hard. As a result, in many works [81, 46, 156, 130], priors are learned over small image patches. This has the advantage of making computational tasks such as learning, inference, and likelihood estimation much easier than working with whole images directly. Surprisingly simple Gaussian Mixture prior (GMM), learned from a set of natural images outperforms all other generic prior methods for image denoising, deblurring, and inpainting. However, its high run time complexity makes it ill-suited for most practical applications.

Markov Random Field (MRF) is another widely used prior to model the statistical properties of natural images [116]. Due to the high run time complexity of GMM, we use MRF due to its simple structure, which only depends on local dependencies and has a small number of parameters [174], which make it suitable for this problem.

Some of the popular MRF image priors realized as regularizers are Gaussian MRF (GMRF), total variation (TV), and Huber MRF (HMRF). We use HMRF as an image prior due to its ability to preserve discontinuities in the images [4].

5.2 Proposed Algorithm

Let X be an image of size $W \times H$ that an adversary wants to contrast enhance. Conventional adaptive CE operation (CLAHE) on image can be represented as a function $\phi(X)$. Conventional enhancement is the direct enhancement operation without any anti-forensic step. To make modified image undetectable to forensic detectors, the process of creating enhanced image should create an image Y which is closer to $\phi(X)$ such that it looks enhanced, but its statistical characteristics are similar to image X. For keeping the statistical characteristics of anti-forensically generated image Y similar to X, we make use of Markov Random Field (MRF) prior which is explained in the following section.

5.2.1 Huber Markov Random Field Prior

The conditional distribution for any pixel $v = (i, j)$ in an image X can be written as:

$$p(X_v | X_{C\text{-}v}) = p(X_{v|N_v}) \tag{5.1}$$

where N_v is the local 8-connected neighborhood at v and $C \in N_v$ is a set of sites in the image. Using the MRF, we can compute the joint probability distribution of a natural image given by Gibbs measure with Huber function as the energy function as [116],

$$p(X) = \frac{1}{z} \exp\left(-\lambda \sum_{c \in \Xi} \rho_t(d_c' X)\right) \tag{5.2}$$

λ defines the degree of smoothness, z is a normalization constant and $\rho_t(.)$ is the Huber function [154].

$$\rho_t(u) = \begin{cases} u^2, & |u| <= t \\ t^2 + 2t(|u| - t), & |u| > t \end{cases} \tag{5.3}$$

where t is a threshold for Huber function as given in [154]. The c in Eq 5.2 is a local group of pixels called cliques and Ξ is the set of all such cliques in X, set Ξ depends on neighborhood structure of HMRF. In this work, neighborhood consisting of the eight nearest neighbors of a pixel is considered. Here, the vectors d_c [154] are chosen to extract the differences between a pixel and its neighbors, such that Eq 5.2 simplifies to Eq 5.4 [158, 139, 159]

$$p(X) = \frac{1}{z} \exp\left(-\lambda \sum_{n=0}^{P-1} \sum_{m \in N_n} \rho_t(X[n] - X[m])\right) \tag{5.4}$$

where N_n is the index set of neighbors for the n^{th} pixel, and P is the number of pixels in the image.

The value of t in Eq 5.2 is determined empirically as described in Section 5.3.2. For maximizing $p(X)$ in Eq 5.2, it is sufficient to minimize the quantity inside the exponential. Hence we include it in our optimization formulation which is explained next.

5.2.2 Optimization Formulation

Optimization formulation for generating anti-forensically enhanced image is given by

$$\begin{aligned} \arg\min_{Y} \frac{\alpha}{2}\|Y - X\|_2^2 + \frac{\beta}{2}\|Y - \phi(X)\|_2^2 + \frac{\Gamma}{2}\left(\sum_{c \in \Xi} \rho_t(d_c^t Y)\right) + \\ \frac{\delta}{2} \sum_{i \in \mathcal{S}} \| M_i \circ D(Y - X)D' \|_2^2 \end{aligned} \tag{5.5}$$

In Eq 5.5, $\|.\|_2^2$ represents square of the l_2-norm and α, β, Γ and δ are regularization parameters. The first and second term in Eq 5.5 reduces the squared distance of anti forensically generated image Y from X and conventionally enhanced image $\phi(X)$ respectively. By varying α and β judiciously, an image close to contrast-enhanced image but not very different from original image in statistical properties can be obtained. The third term in optimization formulation is obtained from Eq 5.2 and calculates sum of squared difference between a pixel and its neighbors in Y and forces output Y to follow HMRF structure. The fourth term in Eq 5.5 minimizes the squared distance between AC coefficients of original image and output image such that characteristics of output image Y are similar to image X in DCT domain. D denotes the block diagonal matrix which has the diagonal blocks of 8x8 DCT matrix. '$\circ$'denotes the Hadamard product. M_i is an indicator matrix given by

$$M_i(k,l) = \begin{cases} 1, & \text{if } (k \ mod \ 8, l \ mod \ 8) \in \mathcal{S}(i), \\ 0, & \text{otherwise.} \end{cases} \tag{5.6}$$

where $\mathcal{S}$ denotes the set of AC coefficients of a DCT coefficient matrix. Set $\mathcal{S}$ consists of six AC frequencies i.e $\mathcal{S} \in \{(0,1),(1,0),(2,0),(1,1),(1,2),(1,3)\}$. High frequencies correspond to rapid changes in pixel intensity over a small displacement (a few pixels). Since most natural images have large features which change relatively slowly, most of the information is in the lower spatial frequency range. High frequencies correspond to noise, sharp edges or very fine textured features. Therefore, we only consider the first six AC frequencies since higher AC frequencies contain less energy. The elements of M_i are 1 for a particular AC component present in set $\mathcal{S}$ while zero for all other components. Though we apply L2 norm for simplicity, an L1 norm based solution can promote sparsity and result in sharper images [18].

5.2.3 Solution to Optimization Formulation

We use gradient descent to solve Eq 5.5. Let $Y^{(i)}$ denote the output at i^{th} iteration, then the gradient descent update can be written as

$$Y^{(i+1)} = Y^{(i)} - \zeta^i \psi(Y^{(i)}) \tag{5.7}$$

where ζ^i is the learning rate. The gradient $\psi(Y)$ of the objective function is given as

$$\psi(Y) = \frac{\alpha}{2}\nabla\phi_1(Y) + \frac{\beta}{2}\nabla\phi_2(Y) + \frac{\Gamma}{2}\nabla\phi_3(Y) + \frac{\delta}{2}\nabla\phi_4(Y) \tag{5.8}$$

where,

$$\nabla\phi_1(Y) = 2(Y - X), \tag{5.9}$$

$$\nabla\phi_2(Y) = 2(Y - \phi(X)) \tag{5.10}$$

$$\nabla\phi_3(Y) = \sum_{c\in C} d_c^t \rho_t'(d_c^t Y) \tag{5.11}$$

$$\rho_t'(d_c^t Y) = \begin{cases} 2u, & |u| <= t, \\ 2t, & u > t, \\ -2t, & u < -t, \end{cases} \tag{5.12}$$

$$\begin{aligned}\nabla\phi_4(Y) &= \nabla \sum_{i\in\mathcal{S}} \parallel M_i \circ \{D(Y-X)D'\} \parallel_2^2 \\ &= \nabla \sum_{i\in\mathcal{S}} [M_i \circ \{D(Y-X)D'\}]'[M_i \circ \{D(Y-X)D'\}] \\ &= \sum_{i\in\mathcal{S}} [M_i \circ \{D(Y-X)D'\}\{M_i' \circ (D'D)'\} + \\ &\qquad M_i \circ \{D(Y-X)D'\}\{M_i \circ D'D\}'] \\ &= 2\sum_{i\in\mathcal{S}} [(M_i \circ \{D(Y-X)D'\})(M_i' \circ (DD'))]\end{aligned} \tag{5.13}$$

$$\nabla\phi_4(Y) = 2\sum_{i\in\mathcal{S}} [(M_i \circ \{D(Y-X)D'\})(M_i' \circ (DD'))] \tag{5.14}$$

We use a constantly decreasing step size ζ starting from 0.1 and reduce at every iteration by a factor of 0.05. This is determined empirically to achieve convergence. The maximum number of iterations is set at 500 and tolerance level for the change in cost is set at 0.0001. Sometimes, the gradient decent takes too many steps to converge and the desired output image may not be obtained. Empirically we find that such cases are rare. Algorithm 1 gives the pseudo-code for the optimization.

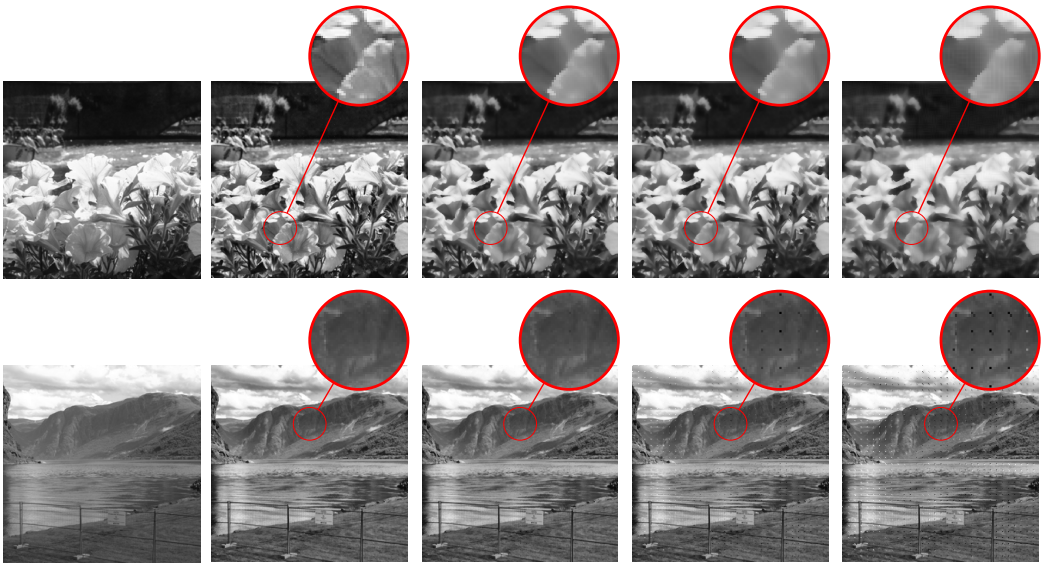


Figure 5.1: ***First row (left to right): Original image, enhanced images using the proposed algorithm for*** $\Gamma = \{0.08, 0.8, 1, 2\}$ ***respectively. Second Row (left to right): Original image, enhanced image for*** $\delta = \{0.5, 1, 5, 10\}$ ***respectively.***

5.3 Experimental Setup and Results

In this section, we first discuss about parameters of the algorithm. Then, we explain the reason for selecting CLAHE over GC and HS as enhancement

operator in Eq 5.5. We then discuss the details of dataset and classifier. Further, we compare the performance with other antiforensic algorithms [11], [27], [151] on the basis of CE detection performance and visual quality. All the experiments with benchmark methods are performed under the same conditions. The comparison between the benchmark and the proposed algorithm is also made on similar grounds.

5.3.1 Parameters

Parameters α, β, Γ and δ in optimization Eq 5.5 control the visual quality of enhanced images. Therefore, the regularization parameters are chosen experimentally to give the best results both in terms of overcoming forensic detectors and visual clarity. The parameter α is associated with first term in objective function which ensures that anti forensically generated image is close to original image. If the value of α is very large then generated image will be close to original image. However, adversary want generated image to be perceptually similar to enhanced image while keeping its statistical properties similar to original image. Hence, large weight cannot be assigned to α.

Similarly, β ensures that the output image Y looks enhanced. Therefore, α and β are varied such that Y has the perceptual characteristics of enhanced image and statistical properties of the natural image. α and β are chosen to be 0.2 and 12 respectively.

Since Γ is associated with HMRF term in Eq 5.5, it promotes smoothness in the image. Fig 5.1 (1^{st}row) shows enhanced images generated for four different values of $\Gamma = \{0.08, 0.8, 1, 2\}$. It can be observed that increasing Γ results in smoother images. Similarly, 2^{nd} row shows enhanced images generated using four different values of $\delta = \{0.5, 1, 5, 10\}$. We can observe that as the value of δ increases, the visual artifacts become prominent. For value of δ less than 1, there are no visual artifacts, however performance of forensics detectors increases for lower value of δ. The value of Γ and δ for our experiments are 0.08 and 1 respectively.

Finally, the value of threshold t is selected according to the references [154, 158, 139, 159]. Threshold t in HMRF image priors is the trade-off between high and low frequency components of image [80]. For small values of t, reconstructed image will have less high frequency components, hence making the image smoother. Whereas higher values of t will increase the high frequency components in image, making it more sharper. Therefore, threshold t in Eq 5.3 is chosen as 10 [154].

Table 5.1: ***Visual quality assessment between antiforensically enhanced images generated using global enhancement techniques such as GC, HS, and CLAHE, where a lower value indicates less distortion. Best results are underlined.***

Image Quality Measure	PSNR	SSIM	MS-SSIM	PSNR-HVS	PSNR-HVS-M	FSIM	MAD	VIF
AntiCE (Proposed)-GC&HS	41.1498	0.9643	0.9558	38.4644	42.8519	0.9797	7.5817	0.8436
AntiCE (Proposed)-CLAHE	39.3143	0.9895	0.9775	39.0128	46.7608	0.9883	0.6729	0.9153

5.3.2 Performance with Global contrast enhancement techniques

We first evaluate performance of the proposed algorithm when GC and HS are used as image enhancement operations. We use GC and HS function as $\phi(.)$ in Eq 5.5. Antiforensically enhanced images are tested for their image quality. In our experiments, we use commonly used image quality measures such as PSNR, Structural Similarity Index (SSIM) [184] and Multi-scale Structural Similarity (MS-SSIM) [183]. Since humans usually use contrast, color, and frequency changes in their measures, we also include Human Visual System-based (HVS) measures such as PSNR-HVS [49], PSNR-HVS-M [145] and Feature Similarity Index (FSIM) [194]. To get the account of distortion introduced in an image due to antiforensics operation, we use Most Apparent Distortion (MAD) index [109]. MAD segregates the distortions present in an image and uses dual strategies for evaluating those distortions. Finally, we compare the conventionally enhanced images with antiforensically enhanced images using Visual Information Fidelity (VIF) [161] index based on natural scene statistics. By using VIF, we can assess the ability of the proposed algorithm in preserving the natural statistics of an image.

Table 5.2: ***Performance analysis for 1^{st} and 2^{nd} order statistics based detectors (FPR = 0.01).***

Detectors	CE [Uniform]	Proposed $\alpha = 0.2$	Proposed $\alpha = 0$
[28]	49.16	50.02	50.45
[48]	73.03	67.70	68.90

Table 5.1 shows comparison of images generated antiforensically when GC, HS or CLAHE are used with the proposed algorithm. It can be observed that the quality of images generated using the proposed algorithm is better when CLAHE is used as an enhancement function. Out of 8 quality measures, CLAHE does better in 7. For example, MAD index for images using GC or HS is 7.5817, whereas, MAD index for CLAHE enhanced images is 0.6729. Also, as explained in Chapter 3, the images generated using adaptive contrast enhancement look more realistic than images generated using global contrast enhancement operations. Since the quality of enhanced

Table 5.3: ***Performance analysis (accuracy in %) for steganalysis features based detectors (FPR = 0.01). CE is performed using CLAHE.***

Method	CE [Uniform]	Proposed $\alpha = 0$	Proposed $\alpha = 0.2$	[11]	[27]	[151]
SRM	99.45	96.03	93.80	94.10	90.10	95.82
SPAM	94.12	76.44	71.76	94.84	63.92	79.86
SPAM-CCPEV	95.97	66.01	63.93	97.55	74.36	96.76

images is an important parameter for antiforensics operations, for further experiments, we use only CLAHE for generating antiforensically enhanced images.

5.3.3 Database and Classifier

We randomly sample 2000 grayscale images from the BOSS dataset [135], crop them from the center to obtain images of size 256×256, thus obtaining never enhanced dataset. We sample 2000 different images from the BOSS dataset and process them in two different ways: CLAHE (with Uniform, Rayleigh and Exponential distribution) and the proposed algorithm. In first case, we enhance the contrast of the images by varying the shape of the histogram according to three different distributions: Uniform, Rayleigh and Exponential. In second case, we apply the proposed anti forensics operation on images enhanced in first case. All the images are saved using JPEG Quality Factor of 90.

We train two-class ensemble classifier proposed in [103] and use accuracy at FPR of 0.01 as a metric for comparing the performance of various detectors. The ensemble classifier consists of many base learners independently trained on a set of never enhanced and enhanced images. Each base learner is a simple random forest classifier built on a (uniformly) randomly selected subspace of the feature space. Given an example from the testing set, the final decision is formed by aggregating the decisions of individual base learners. This supervised ensemble strategy will work only if the individual base learners are sufficiently diverse in the sense that they make different errors on unseen data. To further increase the mutual diversity of the base learners, each learner is trained on a bootstrap sample drawn from the training set rather than on the whole training set. It allows us to obtain an accurate estimate of the testing error, which will be important for determining optimal ensemble parameters.

The feature space dimension for SRM and SPAM features are 34671 and 686 respectively. In case of SPAM & CC-PEV, feature vector has dimension of 960 (686 for SPAM and 274 for CC-PEV without calibration). We perform two different experiments using CLAHE images and images generated by

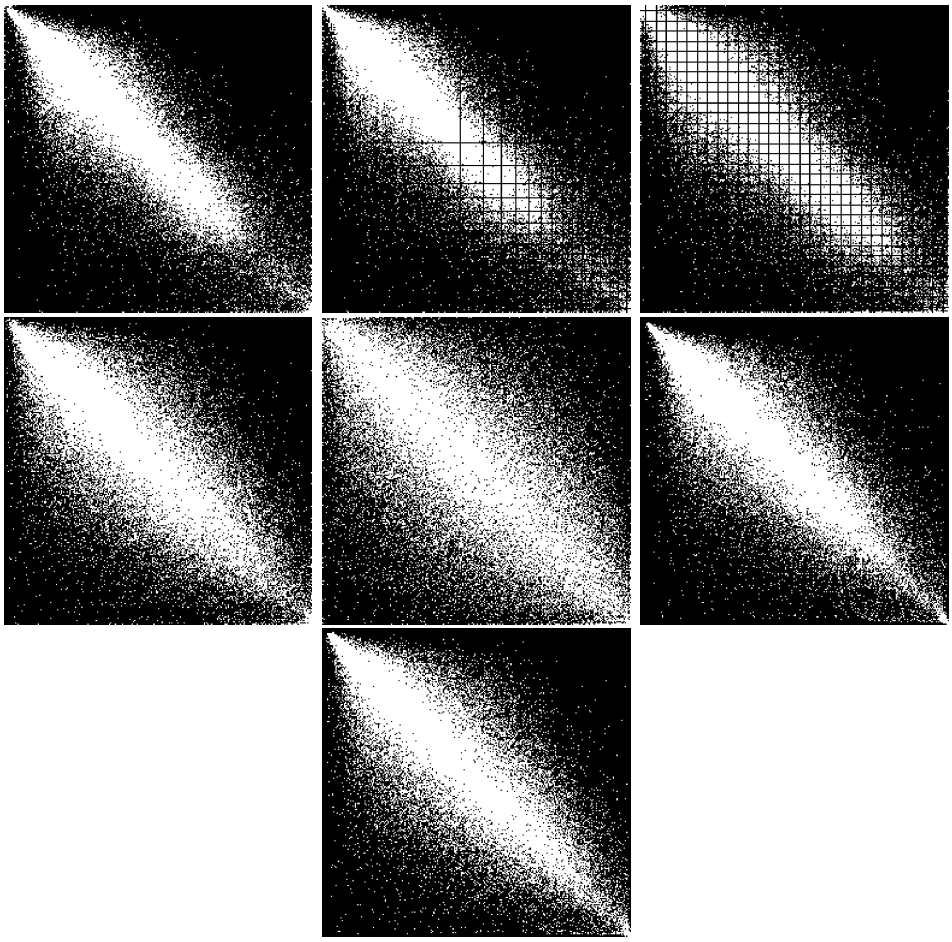


Figure 5.2: ***Input to the neural network proposed in [171] is the grayscale GLCM images of size*** 256×256***. (Top row, left to right) Original image, Gamma Correction (GC), Histogram Stretching (HS), CLAHE. (Bottom row, left to right) Gaussian Dithering (GD), Total Variation (TV) and Proposed.***

the proposed algorithm. In the first experiment, we split the never enhanced dataset and images created using CLAHE into training and testing sets respectively. In the second setup, testing images in first setup are replaced with anti-forensically generated images. Training set is created by randomly selecting 75% of total images in dataset and rest are used for testing. We repeat the process 10 times with different training and testing set, and report the average results.

5.3.4 Performance against CE Detectors

First and Second order statistics detector

The first order forensic detector proposed in [28] detects global contrast enhancement with high accuracy. However, when the image is enhanced using CLAHE, the performance of the detector reduces considerably. From Table 5.2, we can observe that accuracy is 49.16%. Since adaptive contrast

Table 5.4: ***Performance analysis for steganalysis features based detectors for CLAHE using Rayleigh and Exponential distribution for enhancing images.*** $\alpha = 0.2$ ***is used for generating AntiCE images using Eq 5.5.***

Method	CE [Exponential]	AntiCE [Proposed]	CE [Rayleigh]	AntiCE [Proposed]
SRM	99.65	94.69	99.32	93.90
SPAM	90.81	53.34	90.30	54.45
SPAM-CCPEV	90.73	58.32	90.51	57.95

enhancement operates locally, the features extracted using [28] are unable to distinguish between enhanced and original image. Similarly, detection accuracy of first order detector for anti forensically enhanced images using the proposed algorithm is 50.02% and 50.45% for $\alpha = 0.2$ and 0 respectively.

Next, we evaluate performance of the proposed counter forensics technique against second-order statistics features derived from GLCM [48]. From Table 5.2, we can observe that accuracy of GLCM based detector reduces from 73.03% to 67.70% ($\alpha = 0.2$) when testing images are generated using the proposed algorithm.

Steganalysis features based detectors

Spatial domain based detector

Spatial domain steganalysis features such as SRM and SPAM proposed in [72], [143] respectively achieves superior accuracy when compared with first or second order [28], [48] detectors discussed previously. From Table 5.3, we can observe that steganalysis features can detect the CLAHE enhanced images with an accuracy greater than 94% (approximately). Even though enhancement can be detected with high accuracy using steganalysis features, the proposed counter forensics approach reduces the accuracy of CE detectors. For example, in the case of detector using SRM as features, the accuracy of detector reduces from 99.45% to 93.80% when anti forensically generated images are used during testing. Similarly, accuracy for SPAM features reduces from 94.12% to 71.76%.

Joint spatial and DCT domain based detector

The proposed counter forensic technique is also effective in hiding the artifacts generated due to enhancement in both spatial and DCT domain. DCT domain features CC-PEV [144] used along with spatial domain features SPAM [150], [9] has detection accuracy of 95.97% as shown in Table 5.3.

However, it can be observed that accuracy of the detector based on fused features reduces to 63.93% from 95.97%. Similar results can be observed for enhancement methods using Rayleigh and Exponential histogram in Table 5.4. We defer the discussion of comparison results for anti forensics against [11], [27] and [151] in Table 5.3 to Section 5.3.5.

Performance against deep learning based forensic detector

We train the networks proposed in [171] and [160] using GLCM obtained from unaltered, gamma corrected (GC), histogram stretched (HS) and CLAHE images. The training set consists of 4000 images in each category. Thus the total number of images is 16000. Testing is performed for four different cases as explained below

Table 5.5: ***Performance (accuracy in %) against deep learning based forensic detectors.***

Dataset	Case1	Case2	Case3	Case4
[171]	98.75	81.42	79.38	78.58
[160]	99.85	87.01	85.80	82.28

- Case1: Testing set consists of 1000 unaltered, GC, HS and CLAHE images respectively.
- Case2: CLAHE images in case 1 are replaced with anti forensically enhanced images generated using Gaussian Dithering (GD) [27].
- Case3: CLAHE images in case 1 are replaced with anti forensically enhanced images generated using Total Variation (TV) method [151].
- Case4: CLAHE images in case 1 are replaced with anti forensically generated images generated using the proposed algorithm.

In all the cases, the total number of images is 4000. Table 5.5 outlines the results of all four cases. In case 1, we observe that the model proposed in [171] obtains an accuracy of 98.75%. However, we observe a drastic reduction in accuracy in case 2 and case 3. We observe a further drop in the accuracy when images generated using proposed algorithm are tested using our proposed algorithm. For GD [27] and TV [151] methods, accuracy of CNN detector is 81.42% and 79.38% respectively, whereas accuracy for case 4 reduces to 78.58%. We observe similar results for [160].

It is interesting to note that CNN detector can detect conventional CLAHE with high accuracy. However, the accuracy reduces when CLAHE images are replaced with antiforensically enhanced images. The possible reason for such behavior is, the antiforensic algorithm discussed in [11], [27] and [151] try to fill the gaps in GLCM of enhanced images. The inherent approach of

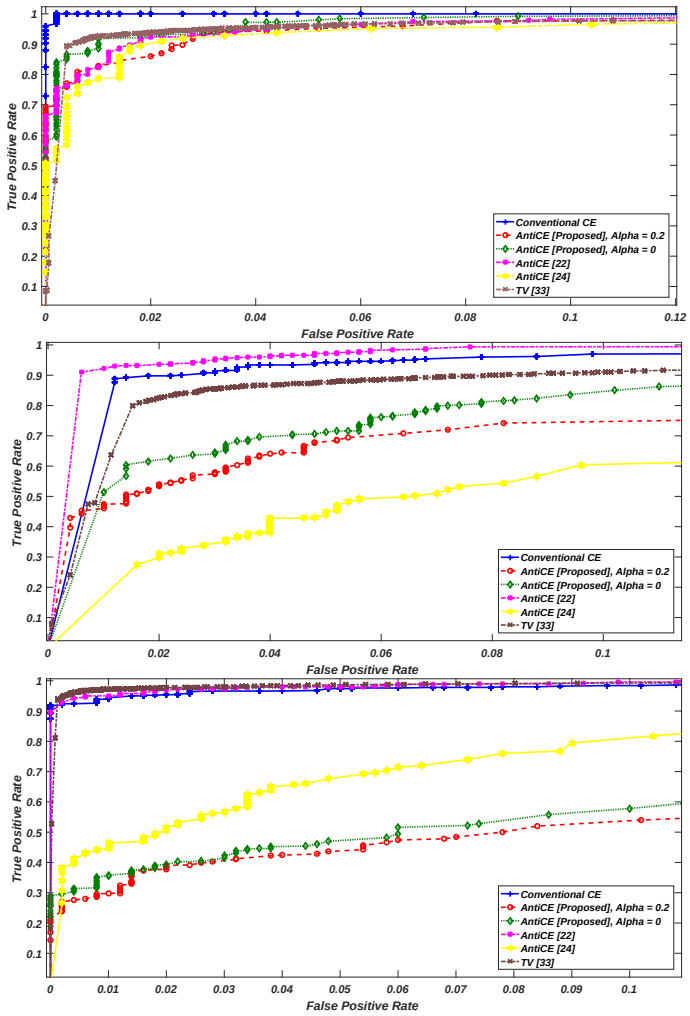


Figure 5.3: ***Performance analysis of forensic detectors based on SRM, SPAM and SPAM-CCPEV feature respectively.***

antiforensic algorithms is to make the GLCM of an enhanced image similar to the unaltered image. Since the input to the detector is GLCM, CNN detector gets confused between GLCM of unaltered images and antiforensically enhanced images which result in the reduction in accuracy. Fig 5.2 shows GLCM of GC, HS, CLAHE, GD, TV and proposed algorithm. We can observe that GLCM of GD, TV and proposed algorithm are very similar to GLCM of unaltered images.

5.3.5 Comparison with other counter forensic techniques

We compare the performance of the proposed algorithm with universal image counter-forensic scheme based on the analysis of the image histogram [11], [27] and [151]. For fair comparison with [11], we substitute value of parameter $\alpha = 0$ in Eq 5.5. Substituting $\alpha = 0$ is similar to the scenario in which the original image is not available to the adversary. We generate enhanced images using [11], [27] and [151] and test the performance

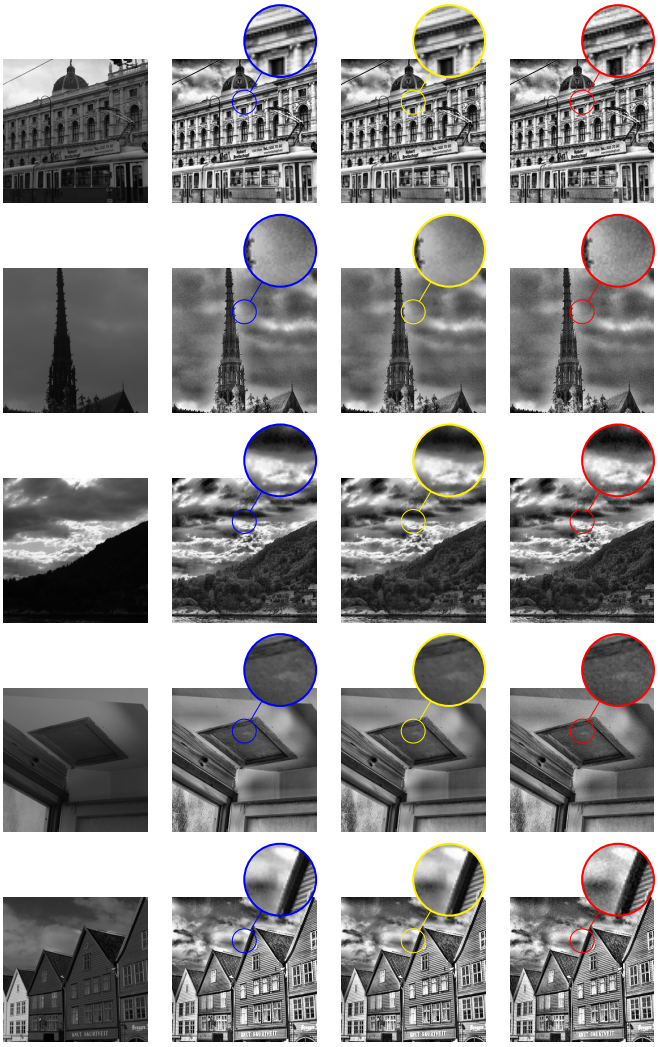


Figure 5.4: ***Examples of images enhanced using CLAHE and images generated anti forensically. (Left to right) Original image, CLAHE enhanced image, proposed and [27]. In magnified portions of enhanced images generated using [27], we can observe visible noise whereas images generated using proposed algorithm are smooth and similar to conventionally enhanced images.***

of steganalysis feature based detectors. From Table 5.3, it can be observed that performance of the proposed algorithm is comparable with universal counter forensic scheme [11] for SRM features. However, the proposed algorithm performs well when spatial and frequency domain features are used for classification. SPAM-CCPEV based forensic detector is able to detect images generated using [11], [27] and [151] with accuracy of 97.55%, 74.36% and 96.76% respectively. Whereas accuracy of same detector is reduced to 63.93% when test images from the proposed algorithm are used. It can also be observed from Fig 5.3 that the proposed algorithm performs better when compared with [11], [151].

Figure 5.5: ***Examples of images enhanced using CLAHE and images generated anti forensically. (Left to right) Original image, CLAHE enhanced image, proposed, [11] and [27].***

Table 5.6: ***Visual quality assessment between antiforensically enhanced images generated using [11], [27], [151] and proposed algorithm, where a lower value indicates less distortion [109]. Best results are underlined.***

Image Quality Measure	PSNR	SSIM	MS-SSIM	PSNR-HVS	PSNR-HVS-M	FSIM	MAD	VIF
[11]	40.9127	0.9497	0.9597	37.5804	40.9169	0.9933	4.7986	0.8717
[27]	47.8000	0.9588	0.9610	38.6087	47.1101	0.9822	2.9125	0.7214
[151]	38.7555	0.9799	0.9744	36.6934	41.1319	0.9873	1.4030	0.8688
Proposed $\alpha = 0.2$	39.3143	0.9895	0.9775	39.0128	46.7608	0.9883	0.6729	0.9153

5.3.6 Visual quality assessment

Another severe challenge for counter-forensics techniques is to keep the visual quality of anti forensically generated image similar to the conventionally enhanced image. We perform a comprehensive evaluation of image quality using image quality assessment measures discussed in Section 5.3.2.

From Table 5.6, it can be observed that out of eight visual quality metrics, proposed algorithm performs better in five. The average PSNR of CLAHE images and images obtained using the proposed scheme is 39.3143 dB which is high and suggests a good visual quality. The proposed algorithm has better SSIM index of 0.9895 in comparison with SSIM of 0.9497, 0.9588 and 0.9744 for [11], [27] and [151] respectively. Also MAD index of the proposed algorithm is 0.6729 which is much lower (better) than 1.4030 ([151]), 2.9125 ([27]) and 4.7986 ([11]). A similar observation can be made for other quality measures (MS-SSIM, PSNR-HSV, VIF). In addition, if we magnify the images from different anti-forensic algorithms, we can observe that the proposed algorithm has a better visual image quality. In Fig 5.4, the magnified portion of enhanced images show noise artefacts, whereas magnified view of images generated using the proposed algorithm is better and close to conventionally enhanced images.

5.4 Discussion

From results described in Section 5.3, it is evident that the proposed anti-forensic approach can achieve the goal of generating images which are visually convincing as well as less likely to be detectable by CE detectors. Fig 5.5 shows few examples of images enhanced using conventional enhancement method, proposed algorithm, [11] and [27]. It can be observed that images generated using the proposed framework are similar to conventional CE images and have better image quality than [11], [27] as discussed in section IV.

Another aspect to note is that all the CE detectors use some form of spatial relationship of pixel or its distribution. By adding the HMRF prior term in the formulation, we can enhance the image without significantly altering the original HMRF structure of the natural image. HMRF term ensures that the structure of anti forensically modified image is not much different from CE image and also not far from the natural image. Hence, the proposed algorithm can find the middle ground between natural and contrast-enhanced image such that visually and statistically image is CE but able to deceive the detectors. Although we use HMRF prior to promote smoothness, other priors of similar nature such as Gaussian Mixture Model [152] or Fields of Experts [155] can also be used.

If we compare the anti-forensic algorithms in their ability to reduce the CE detection rate, we can observe that [27] performs better than the proposed algorithm in the spatial domain. However, the proposed algorithm has better performance than [27] if combined features (SPAM-CCPEV) from spatial and DCT domain are used for classification. Also, regarding visual quality, the proposed algorithm has an edge over other anti-forensic algorithms which is evident from Fig 5.4 and Table 5.6.

At this point, we have already discussed two major problems in multimedia forensics, the first one is related to forensics analysis of images for determining their source and the second one is counter forensic of adaptive contrast enhancement. In next chapter, we present the new research area of Egocentric analysis of dash-cam videos, where we will discuss algorithm for linking a dash-cam video to a specific car.

6 Egocentric analysis of Dashcam videos

In this chapter, we propose an algorithm for linking a dashcam video to a specific car in Section 6.1. We further investigate the process of forging the signature of a car in Section 6.3 and propose a counter forensics method to detect such forgery. We also discuss the application of our technique to other potential platforms in Section 6.4.

6.1 Proposed Algorithm

Our algorithm is motivated by the fact that due to the motion of a vehicle and the vibrations generated by its engine, the video recorded by a dashcam will be degraded by motion blur. The proposed algorithm can be divided into two stages. In the first stage, we extract the blur kernel from each frame of the video sequence. In the second stage, we extract features from extracted blur kernels and train a classifier for the identification of the vehicle.

6.1.1 Blur extraction algorithm

Blind deconvolution techniques discussed in [68], [105], [172] can be used for extracting a motion blur kernel due to vehicle motion. Blind image deconvolution recovers a sharp image from a blurry and noisy image. Since both the blur kernel and sharp image are unknown, it is considered an ill-posed problem. We use the standard model for a blurred image as proposed in various blind image deconvolution algorithms. Since we are working with videos, henceforth, we will consider video frames in place of images. The frame degradation model is given by

$$F = KX + N \tag{6.1}$$

where F is observed frame with blur, X is the unknown sharp frame, K is the blur kernel matrix, and N is the additive Gaussian i.i.d noise.

The estimation of blur from the frames captured by a moving camera is quite challenging. Linear motion and vibrations due to the engine rotates

the camera around the Z-axis (axis perpendicular to the plane of the camera sensor) because of which spatially variant blur is generated in every frame captured by the camera. Since blur is not spatially constant, spatially variant blur can be obtained using the method proposed by Krishnan *et al.* [105].

Estimation is performed on high frequencies of frame. For estimating the blur kernel, discrete filters $\nabla_x = [1, -1]$ and $\nabla_y = [1, -1]^T$ are applied to convert the given blurry and noisy observed frame F to a high frequency frame $Q = [\nabla_x F, \nabla_y F]$ and unknown sharp frame X to a high frequency frame, $Y = [\nabla_x X, \nabla_y X]$. The cost function for spatially variant blurring is:

$$\min_{Y,K} \lambda \|KY - Q\|_2^2 + \frac{\|Y\|_1}{\|Y\|_2} + \varphi \|K\|_1 \tag{6.2}$$

subject to constraints $K \geq 0$ and $\sum_i K_i = 1$. Note that Y is the unknown sharp frame in high frequency, K is the unknown blurring kernel. The first term in Eq. 6.2 accounts for the model in Eq. 6.1. The second term is a regularizer on Y. The third term is added to reduce noise in the kernel K (sum to 1 and nonnegative). The scale weight λ and φ control the relative strengths of the kernel and image regularization terms respectively. The non-negativity and sum-to-1 constraints on blur kernel K follow from the principles of blur formation. Due to the $l_1 - norm/l_2 - norm$ regularization term, the optimization problem given in Eq. 6.2 is non-convex. To solve such problems, variables to be minimized i.e Y and K are initialized first, and then the final solution is obtained by alternating between Y and K updates We use the algorithm presented in [105] to obtain Y and K[1].

6.1.2 Feature Extraction

After blur is extracted from each frame, each blur kernel K is vectorized to obtain a vector of 961×1 (since the size of the blur kernel used is 31×31). To capture the change in each element of the blur kernel, we vectorize feature vectors column-wise and stack them along columns to form a blur sequence, as shown in Fig 6.1. Due to motion and vibrations produced by the engine of the vehicle, blur kernels are non-stationary. This means that their frequency-domain representation (their spectrum) changes over time. For capturing this change, we use the Short Time Fourier Transform (STFT). We extract signal using the window of size W and analyze it further using DFT. We use a shift of size s. Thus, frequency components can be used as unique features that can describe the change in the spectrum and can also be uniquely associated with the vehicle.

In the first step, a window of size W is used to extract a matrix $\hat{X}$ of size $961 \times W$ from the blur sequence. After obtaining the matrix $\hat{X}$, Fast Fourier Transform (FFT) is performed along the rows of $\hat{X}$ as shown in Fig 6.1. Here,

[1]Detailed algorithm for blur extraction is given in Appendix B

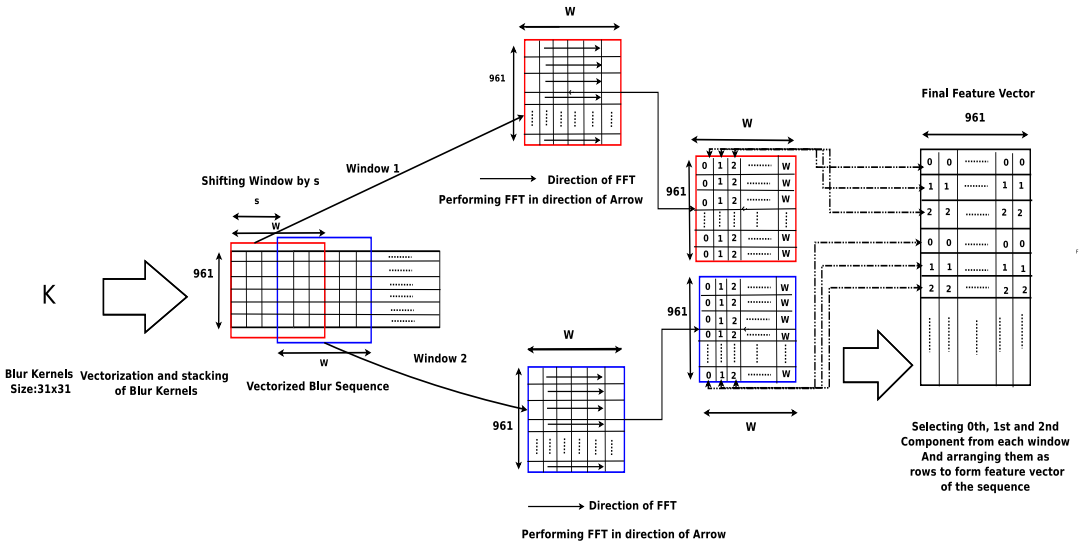


Figure 6.1: ***Figure explains feature extraction process from blur sequence for two consecutive windows. Window of size W is shifted by s and DFT is performed along the rows of the extracted window. $0, 1, 2....W$ denote frequency indices. The feature vector is created by stacking frequency components as shown in figure. (Best viewed in color)***

the elements in each row correspond to a blur kernel sample across time. Thus, FFT along rows captures the variations in blur kernel across time. In the second step, the window is shifted by s number of columns, and the same operation is repeated until the end of the blur sequence is reached. These changes in blur can be periodic due to the combustion and exhaust cycle of the engine and also due to uniqueness in the motion of a vehicle. To localize such variations, we use several overlapping windows. Selecting overlapping windows not only helps in capturing variations in the specific interval of time but also captures how the blur sequence changes when we progress in time. It should be noted that the size of each frequency component is 1×961, and after stacking from each window, 3×961 features are generated, which can be visualized from Fig 6.1. We empirically determine the window size, shift and number of frequency coefficients. We visualize the extracted features using t-stochastic neighbor embedding (t-sne) [128].

Fig 6.2 shows a feature plot for four different cars using t-sne. Each point represents the higher dimensional feature in the 2D plane. We can observe that frequency features from the same cars are clustered together, whereas a considerable margin separates features from different cars.

6.1.3 Classifier

We use random forest [22] for classification and to determine the similarity between videos. The performance of random forest classifier can be further improved by using AdaBoosting. When used with decision tree learning, information gathered at each stage of the AdaBoost algorithm about the

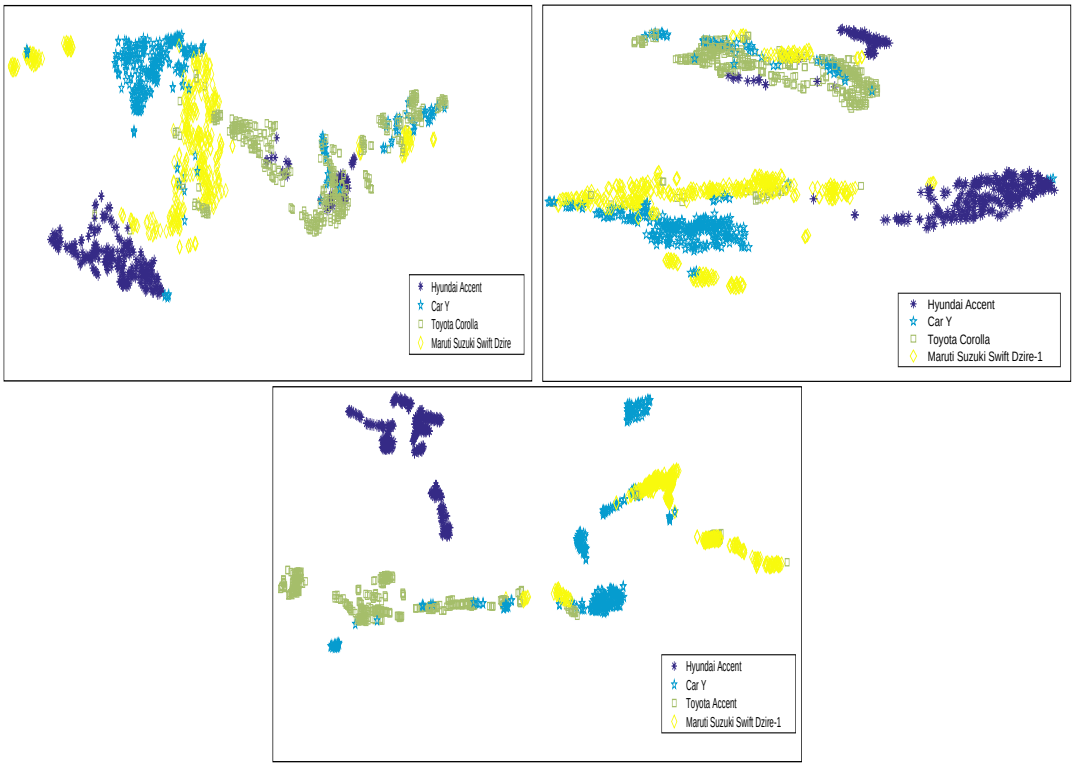


Figure 6.2: 0^{th} ***(top),*** 1^{st} ***(middle) and*** 2^{nd} ***(bottom) frequency feature plot using t-sne for 4 different cars. We can observe the clustering behavior for all three frequency components from same car. Also features from different cars are separated by considerable margin (Please refer to web version for better visualization).***

relative 'hardness' of each training sample is fed into the tree growing algorithm. Later during classification process, trees can use such information to focus on harder-to-classify examples.

Decision Tree

A decision tree is a binary tree with nodes corresponding to the attributes in the input feature vector. At nodes, it is decided how to propagate the given attribute set down the tree. For classifying the given set of features in a particular class, each feature vector from the set is propagated down the decision tree, and the decision is made based on the terminal node that is reached. Decision trees are grown using a training set.

Random Forest

A Random Forest uses a collection of multiple independent decision trees for classification and other tasks (calculating similarity in our case). The final prediction of the class can be made based on the majority voting on the decisions of its constituent decision trees.

AdaBoosting

The accuracy of the classifier can be further improved by using AdaBoosting. AdaBoosting combines multiple weak classifiers into a single strong classifier. Each weak classifier is trained on a subset of the total training set. The data points in the training set are assigned weights, which are equivalent to the probability with which it should appear in the training set. Data points with higher probability are more likely to be included in the training set. After training a classifier, AdaBoost increases the weights of miss classified points so that they can be used for creating weak classifiers. Once all the weak classifiers are created, different weights are given to different classifiers based on their accuracy. More accurate classifiers are given higher weight. Further, all the weak classifiers are combined using equation: $H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)$. Here T is the number of classifiers, α_t is the weight on classifiers and $h_t(x)$ is the output of the weak classifier t . The output is a linear combination of all the weak classifiers, and the final decision is made by observing $H(x)$.

6.1.4 Labeling Strategy

Outputs of random forest classifier are the predicted labels associated with each feature vector. Since our task is to perform classification of dashcam videos, we propose to use thresholding and majority voting over predicted labels for assigning a particular class to the test video.

Thresholding

For a particular video sequence, we can compute the distribution of feature vectors in various classes using labels predicted by random forest classifier. If the percentage of feature vectors classified under a certain class exceeds the predefined threshold, we assign the label of that class to the video. However, it is possible that the threshold is crossed by multiple classes. In such a case, we choose the label of the class, which has a maximum percentage of feature vectors.

When the number of cars present in the database increases, finding an appropriate threshold is difficult. Therefore, we propose an alternative strategy for this scenario.

Majority Voting

In majority voting, we assign the label of a class to the video sequence, which has a maximum percentage of feature vectors classified under it. The advantage of using majority voting is that it reduces the search space to only

a few classes, which can be very helpful in many forensic scenarios such as vehicle forensics. In most of the forensic cases, we do not need to search for evidence in a large domain, hence reducing the search space will help in speeding up the investigation process.

6.1.5 Random Forest Similarity

In many practical scenarios, we have access to the source of evidence. For example, in the source camera linking problem [125],[43], we have access to the camera, and we have to prove that a particular image is captured by the given camera. Similarly, in vehicle forensics, it is assumed that we have access to the vehicle, and we have to answer the question *"Is video under investigation captured using the given vehicle?"*. For such cases, we propose a random forest similarity measure [148]. From a generated random forest using training data, the similarity between two video sequences V_1 and V_2 can be computed as follows. For each video, features are extracted, as explained in Section 6.1.2. Let F_1 and F_2 denote the set of features extracted from V_1 and V_2 respectively. Each feature vector from set F_1 and F_2 is propagated down within all the trees in a random forest. Next, the terminal node position of all feature vector in a set for each of the trees is recorded. Let Z_{ij} denote the terminal node position of i^{th} feature vectors in j^{th} tree. Then similarity between V_1 and V_2 is given by

$$S = \frac{1}{n} \sum_{j=1}^{k} \sum_{i=1}^{m} \mathbb{1}(Z_{ij}^{(1)} = Z_{ij}^{(2)}) \tag{6.3}$$

where superscript l in $Z_{ij}^{(l)}$ denotes set number, $\mathbb{1}$ is the indicator function, n is number of trees in random forest and m is number of feature vectors in feature set (without losing generality we assume that the number of frames in videos to be compared are same, this leads to same number of features in each feature set F_1 and F_2).

Fig 6.3 explains the above procedure using toy example with $m = 3$ feature vectors and $n = 4$ decision trees. Terminal node positions for 1^{st} frequency in different decision trees for feature vector $F1$ are 1, 5, 2, 3 respectively whereas these values for feature vector $F2$ are 1, 2, 5, 3. Similarly, terminal values for every feature vector in a video sequence is computed. Finally, we compare the obtained values with each other and compute the similarity using Eq 6.3.

We use classifier and similarity score as calculated above for determining the source of the video and for forgery detection. Section 6.1.3 gives more details about parameters used for classification and calculating similarity.

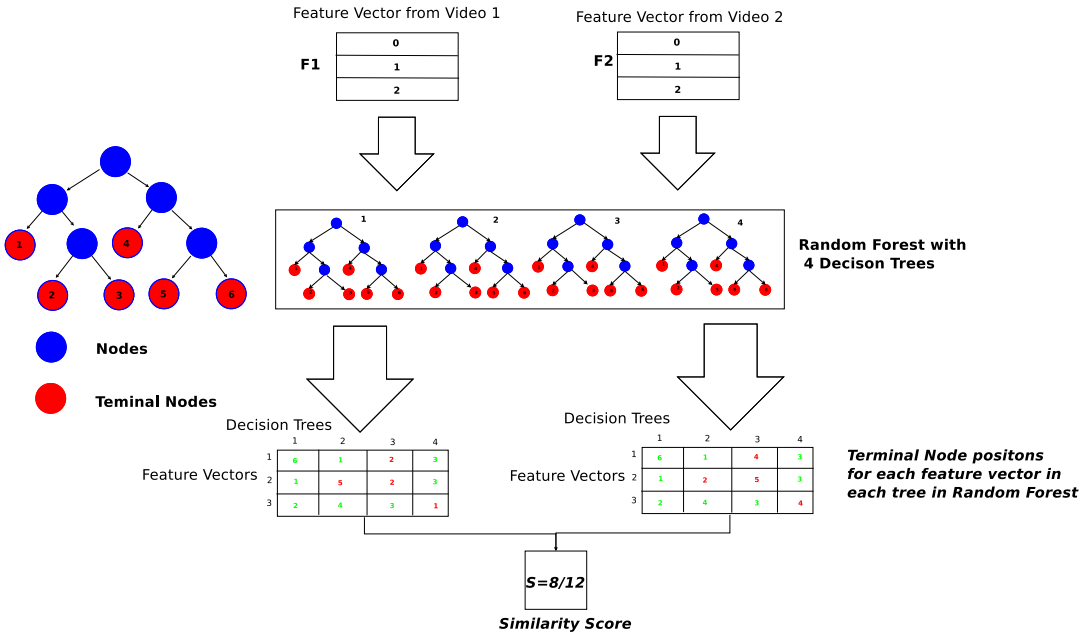


Figure 6.3: ***Similarity calculation using random forest. Numbers in green represents same terminal node position whereas red signifies different. Out of*** 12 ***positions,*** 8 ***are similar where as*** 4 ***are different. Therefore*** S ***is equal to*** 2 ***using Eq. 6.3.***

6.2 Dataset

Database involves videos recorded by 28 different people consisting of university students, professors and employees. There are a total of 25 cars involved out of which we collected data using 23 and downloaded data for the remaining two cars from YouTube. For a particular car, approximately 45 minutes of video is recorded under different conditions (mentioned later in this section). A time gap of 2-3 months is maintained while recording video for the same car. The complete process of video acquisition is done over a span of 10 months. Since recorded videos also consist of unwanted or corrupted frames, we clean the database manually before using the videos for experiments. Some cars used for collecting videos are of the same model as Maruti Suzuki Swift Dzire and Hyundai Grand i10. Each car has a minimum 6000 frames of data available for training and testing. Videos are captured at different places and in a varied environment. Table 6.3 lists the car models used in experiments. All the recorded videos have a resolution of at least 720p, whereas videos downloaded from YouTube have a resolution of 480p. Poor resolution videos are not used for experiments. Details of mobile phones used for capturing the video is given in Table 6.1. Some mobile phones (iPhone 5S and 6S) used in recording the videos also have an inbuilt video stabilization feature. In the case of Youtube[2] videos, we choose only two as these are videos of the reasonable resolution, of sufficient length and captured on different days. Since we are not aware of the model of these

[2]https://www.youtube.com/user/sgrecklessTV

Table 6.1: ***List of mobile phones used for capturing data. Apple iphone 6s and 5s have inbuilt video stabilization.***

Mobile Name	fps	Resolution
Motorola X Play	1080p@30fps	1080x1920
HTC Desire 10 Pro	1080p@30fps	1080x1920
One plus 3	1080p@30fps	1920x1080
Apple iPhone 5S	1080p@30fps, 720p@120fps	640x1136
Apple iphone 6S	2160p@30fps, 1080p@60fps, 1080p@120fps, 720p@240fps	1334x750

cars, we will refer to these as Car X and Car Y from here on. Since blur estimation is highly dependent on lighting conditions [88], we record the data during period of the day when sufficient light is available (for example, in morning, afternoon or in evening before sunset).

Figure 6.4: ***Database.***

To demonstrate that our algorithm can capture the dynamics of a moving vehicle accurately, we create the database of videos captured in the following varied conditions.

Different Locations: Database consists of videos captured on roads in Singapore and India. Since there is a considerable difference in the traffic conditions of Singapore and India, it allows us to study the effect of change in location.

Different days and time: Extracted features should be independent of day and time of the video acquisition. Therefore, videos are captured on different days and time.

To add more variance to the database and to show that unique signatures in videos captured by dashcam is generated due to subtle motion of a vehicle and not because of camera used, we captured videos with the following conditions.

Different Cameras but Same Vehicle: Different dashcams are used to record video from the same vehicle to identify the effect of changing the dashcam in a given vehicle.

Different Vehicles but Same Camera: The same dashcam is used to capture videos from different vehicles in order to show that the signature captured by our algorithm is unique and is dependent only on the vehicle in which the camera is mounted.

Driven by Different Drivers: We collect data of some cars driven by multiple drivers. We create this set to study the impact of driving behavior on the signatures. We randomly let the different drivers drive same car.

6.2.1 Training and Testing Set

We split the videos recorded by each car into a smaller duration video sequence for creating training and testing sets. The length of each sequence is approximately 10 - 15 seconds. Also, there is a difference of at least 200 frames between any two video sequences from the same class (for simplicity, we keep the length of all video sequences same). After splitting operation is over, each class has 12 video sequences of a smaller duration. Therefore, there are 300 such videos for 25 classes. From each class, we use 50% of videos for training, 10% of videos for creating reference set, which is used for computing similarity whereas remaining are used for testing. We repeat the process ten times with different training and testing sets. The results of each iteration are averaged to obtain average accuracy for each class.

6.3 Experiments Setup and Results

6.3.1 Experimental Set Up

We test the performance of the proposed algorithm under various conditions. The signature generated due to the motion and vibration of a vehicle can depend on its velocity and condition of roads. Therefore for training and testing, we use the algorithm explained in Section 6.1.1 for extracting blur kernel from each frame. It is observed that the performance of the algorithm is sensitive to the size of the kernel. Hence, the appropriate size of the kernel is determined experimentally. Large blur kernel gives better classification accuracy, which is expected since larger kernel captures the motion more accurately. However, estimation of large kernel is computationally expensive. Therefore, size of the kernel should be selected judiciously by considering

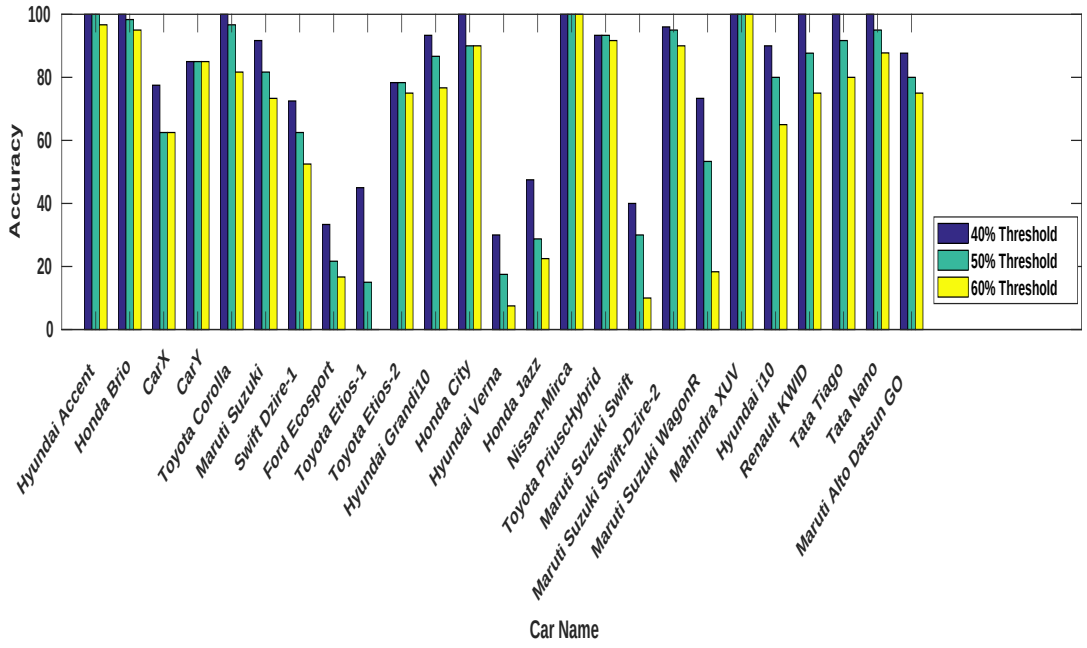


Figure 6.5: ***Classification accuracy for three different thresholds (in %)***

trade-off between the size and computation time required to compute the kernel. The experiments are conducted with $15 \times 15, 21 \times 21, 25 \times 25$ and 31×31 size kernels. The 31×31 size kernel gives better accuracy and is computationally inexpensive. Hence it is preferred over the other kernel sizes. We choose W and s as 5 and 2 respectively.

Blur kernels extracted from 6 consecutive frames for 3 different cars are shown in Fig 6.6. It can observe that blur kernels from the same car shows similar pattern. However, the pattern observed across different cars is significantly different.

To evaluate the performance of our proposed algorithm, elaborate experiments are performed using the following two approaches: (a) Classification using random forest and (b) Predicting label using random forest similarity. For predicting labels using classifier, we use RF classifier and labeling strategies as discussed in Section 6.1.3 and 6.1.4. For computing similarity score, the procedure is described in Section 6.1.5. The dataset is divided into training and testing set as discussed in Section 6.2.1. The number of decision trees n in random forest is decided experimentally. From the experiments, it is observed that the larger the value of n, better is the prediction, but computational cost increases with large value of n. Hence, we choose n such that prediction rate is high while keeping the computational cost low. We choose n as 450 in all our experiments. For calculating similarity using random forest, the training set used for generating the forest is the same as the one used for classification. Also, while calculating the similarity, a reference video is required whose source of origin is known. Therefore, from the dataset, we choose one reference video per class which is not included in training or testing set, as explained in Section 6.2.1.

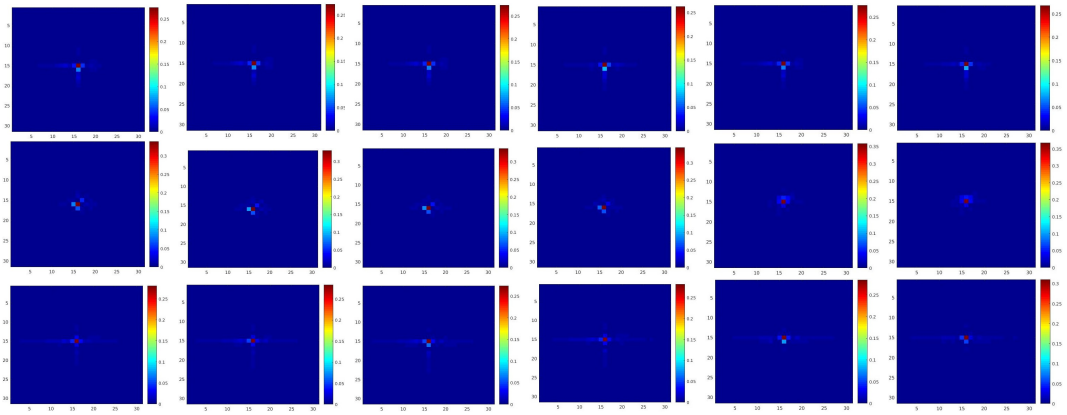


Figure 6.6: ***Blur kernel extracted from 6 consecutive frames for Honda city (top row), XUV (middle row) and Swift (last row) (Best viewed in color).***

6.3.2 Classification Results

The training set is used to generate random forest with 450 decision trees. Each feature vector from the testing set video clips is passed through random forest and the corresponding label is predicted. It is not necessary to correctly classify all the features extracted from the video clip. The decision regarding label of the clip is made collectively using labels of all the features extracted from that particular clip. Therefore, the total number of feature vectors created for one clip will depend on the number of times window W is applied over blur kernel sequence. We first use thresholding for labeling the videos. We perform experiments for the threshold of 40%, 50% and 60%. We can observe from Fig. 6.5 that average accuracy for different classes decreases as the threshold increases, which is expected as increasing threshold puts stricter constraint for labeling the video. The average accuracy for 60% threshold is 75.33% which is still reasonable. It should be noted that false positives under 60% threshold are still under 1%. However, false positives increase to 5% when we decrease the threshold. At 50% threshold, false positives are 2% with the true positive rate of 83.33%.

Thresholding gives us an estimate of how many features per class are correctly classified. A disadvantage of thresholding is that for some videos correct class can have maximum correctly labeled features, however the count fails to cross the threshold. Also, when size of the databset is increased, thresholding approach can lead to incorrect results. To solve problems associated with thresholding, majority voting can be used as labeling strategy.

Using majority voting, we can compute rank based metric commonly known as Cumulative Match Characteristic (CMC). CMC is popularly used in face recognition [89], [33], Person re-identification [140], [195], [127] and in biometric evaluation and testing [133]. For generating CMC, each video clip is compared against all training video clips and resulting scores are sorted and ranked. The resulting score can be used for determining the rank at which a true match occurs. The class obtaining the maximum number of features is ranked 1, second maximum is ranked 2 and so on. This process is repeated

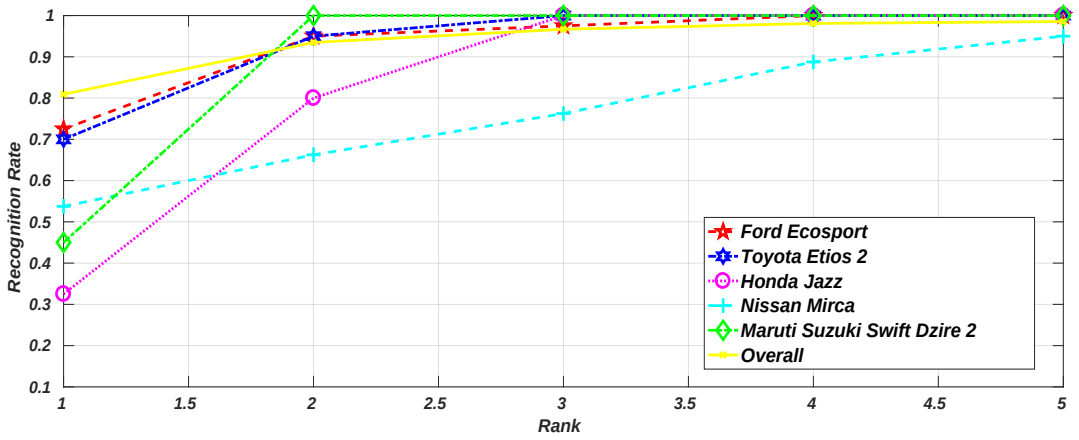


Figure 6.7: ***CMC curve for average accuracy over 10 fold cross validation.***

for all the videos. Table 6.3 shows rank 1 to rank 5 accuracy for each class. From Table 6.3, we can observe that rank 2 accuracy increases from 79.68% to 91.76%. Hence, the proposed algorithm can link the video to a particular car with considerably high accuracy. From Table 6.3, it can be observed that the accuracy for certain cars is low. For example, Toyota Etios-1 and Honda Jazz rank 1 accuracy is 32.33% and 30.5% respectively. Possible reasons can be the poor quality of roads under which the video is recorded using these cars. Fig. 6.7 shows the CMC curve for 5 different cars and overall average accuracy.

Performance of different classifier

We compare performance of Multi Class SVM (MSVM) [110] and Naive Bayes (NB) classifier with random forest. Table 6.2 shows overall accuracy for different classifier. For threshold of 50%, MSVM and NB have average accuracies of 65.30% and 73.46% respectively, whereas RF with adaboost has average accuracy of 85.62%. We can also observe that RF-AdaBoost shows the best performance for all thresholds. AdaBoosting improves the performance of RF classifier by approximately 2%.

Table 6.2: ***Performance of different classifier. Random forest (RF) performed better than other classifiers. AdaBoosting further improved the performance by approximately 2%.***

Classifier	40%	50%	60%
MSVM	73.46	65.30	55.10
NB	81.63	73.46	61.22
RF	89.31	82.92	75.50
RF-AdaBoost	***92.83***	***85.62***	***77.02***

Table 6.3: ***Rank 1-5 accuracy for 25 cars (in %) and classification accuracy for three different thresholds. (in %)***

Car Model (labels)	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
Hyundai Accent (1)	100	100	100	100	100
Honda Brio (2)	100	100	100	100	100
CarX (3)	70	81	85.50	91.50	95.33
CarY (4)	82	82	84	85	85
Toyota Corolla (5)	95.33	99	99	99	100
Maruti Suzuki Swift Dzire-1 (6)	92.33	97.33	97.33	98.33	100
Ford Ecosport (7)	71.50	91	97.50	98	100
Toyota Etios-1 (8)	32.33	59	81.33	91.33	94.97
Toyota Etios-2 (9)	69	94	99	99	100
Hyundai Grandi10 (10)	77.33	100	100	100	100
Honda City (11)	94.67	100	100	100	100
Hyundai Verna (12)	99	100	100	100	100
Honda Jazz (13)	30.50	76	100	100	100
Nissan-Micra (14)	51.75	62.25	74.25	86.75	95
Toyota PriuscHybrid (15)	98	99	99	100	100
Maruti Suzuki Swift (16)	93.33	95.67	94.67	94.67	96.67
Maruti Suzuki Swift-Dzire-2 (17)	42	97	98	100	100
Maruti Suzuki WagonR (18)	94	99	100	100	100
Mahindra XUV (19)	83	94.67	100	100	100
Hyundai i10 (20)	100	100	100	100	100
Renault KWID (21)	70	91.33	99	99	100
Tata Tiago (22)	72.50	94.50	99	100	100
Tata Nano (23)	92.33	96.33	99	99	100
Maruti Alto (24)	98	99	100	100	100
Datsun GO (25)	83	86	92.50	95.50	100
Average	**79.68**	**91.76**	**95.96**	**97.48**	**98.67**

Different window size and shift

As discussed in previous section, for capturing change in blur sequence with time, window size and shift plays significant role. Hence, it is crucial to determine their optimal values. We perform experiments with different window size and shift for determining their effect on performance. Fig 6.8 shows ROC for different W and s. From Fig 6.8, we can observe that size of window W should not be very large since large window won't be able to localize the unique features in blur sequence. Window size of 5 performs better than 15 and 25. Also, extracting features from overlapping window results in better performance. Window size of 5 and shift of 2 has better performance than window size of 5 and non overlapping windows.

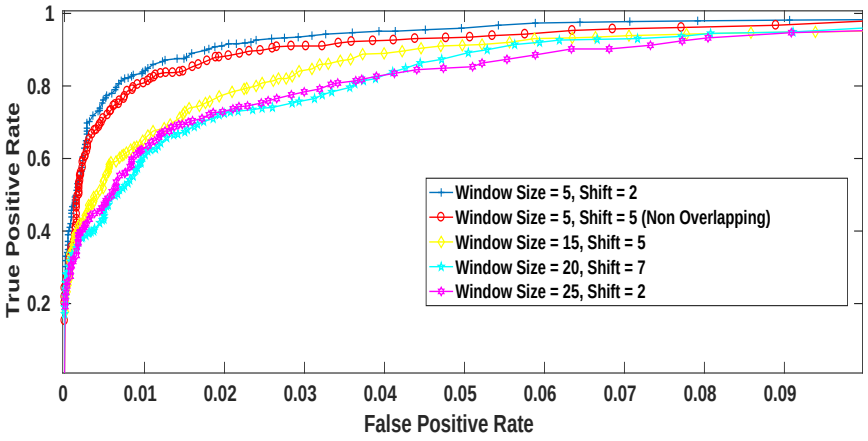


Figure 6.8: ***ROC plot for different window size and shift. Window size = 5 and shift = 2 has better performance. Therefore,*** $W = 5$ ***and*** $s = 2$ ***are used for creating feature vectors in all experiments.***

Varying frequency components in feature vectors

In our experiments, features are created by using 0^{th}, 1^{st} and 2^{nd} frequency components. We observe that there is negligible effect on the performance if we decide to select more frequency components for creating the feature vector. Fig 6.9 shows ROC curves for features created using different number of frequency components. We can observe that selecting the 20 AC components have negligible effect on the performance. However, if we further increase the number of frequency components in the feature vector, performance starts to degrade (Fig 6.9). The decrease in performance due to large number of frequency components is due to the presence of noise in the higher frequency. There are various sources which can add noise to the blur kernels. For example, due to the uneven roads dashcam can experience high jitters which are absent when the road is smooth. Such type of noise produce higher than normal changes in the blur, hence can be considered as high frequency. To reduce the effect of the noise, extracted features are created using the low frequency components only.

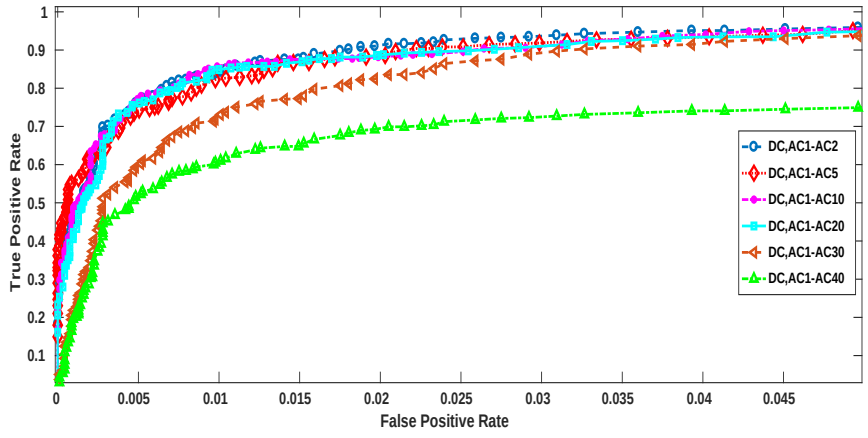


Figure 6.9: ***ROC plot for varying frequency components in feature vectors.***

Also, size of the feature vector is directly proportional to the number of frequency components selected for creating the feature vector. Therefore, selecting more frequency components will only increase the computational cost. Hence, we use DC, AC1-AC2 to construct the final features.

Effect of change in light conditions

To investigate the effect of lighting conditions, we try to keep the road and scene content fixed. We use 5 vehicles towards this. We collect the data in university campus which gives us better control over scene content. We follow the same path for capturing the video for all vehicles. Data is recorded during the early morning and afternoon when there is negligible traffic on the campus roads. Lighting conditions also vary significantly during these hours. We also use different cameras to record the data. Table 6.4 details performance of the proposed algorithm for different conditions. We can observe from the first two columns that changing light conditions does not have any effect on the performance of the proposed algorithm. Although there is significant difference in light conditions during early morning and in afternoon, sufficient light is available for blur estimation algorithm to give a good estimate of blur kernels. From first and third column of Table 6.4, we can also observe that using different cameras have negligible effect on performance of the proposed algorithm. Also, from column three of Table 6.4, we can infer that if light conditions are similar, the proposed algorithm gives optimal results.

Table 6.4: ***Accuracy of the proposed algorithm (in %) under different conditions using RF-Adaboot classifier with majority voting. (A:Same Road, B:Same Camera, C:Same scene content, D:Same light Condition)***

Car Name	A,C	A,B,C	A,C,D
Hyundai Grandi10	75	75	100
Maruti Suzuki WagonR	100	100	75
Volkswagen Vento	100	100	100
Nissan Micra	50	50	50
Toyota Corolla	100	100	100

Effect on performance due to driving style

From classification results in Section 6.3.2, it is observed that Toyota Etios-2 and Nissan Micra have recognition rate of 15% and 28.75% for 50% threshold. Both of the cars have videos recorded by six different drivers. Therefore,

possible reason for poor performance can be the different style of driving of different persons. Studying the effect of driving style on the proposed algorithm is a complex problem and requires in depth analysis on much larger database which can be further investigated in future work.

6.3.3 Using random forest similarity for prediction

Thresholding or majority voting may not be good strategies when there is a possibility of forgery. To this end, we compute similarity using the algorithm described in Section 6.1.5 between the testing and reference dataset. Random forest is generated using the training dataset created for classification problem (Section 6.2.1). Table 6.5 shows the similarity score calculated using Eq. 6.3 between four videos and reference videos. Similarity ratio r is computed for each class using the equation

$$r = \frac{S_{max}}{S_i} \tag{6.4}$$

$\forall i = 1, 2, ...25$, where S_{max} is the maximum similarity score for a particular video and S_i is the similarity score for i^{th} class.

We decide whether test video is similar to reference video if this ratio is less than predetermined threshold τ. A similar procedure is used in Scale Invariant Feature Transform (SIFT) [123] for eliminating the false matches. In Table 6.5, for each video, ratio column is calculated using Eq. 6.4. Threshold is decided empirically to give best results. Out of 75 test videos, we are able to predict correct labels for 48 videos using random forest similarity. For rest of the videos, this procedure predicts more than one label. We detail the results for two best and two worst performing cases in Table 6.5. We can observe that labels for Video1 and Video2 are predicted correctly, but for Video3 and Video4 we have multiple labels. However, search space in both of the cases reduces to few cars. It should be noted that for Video3 and Video4 predicted labels have ground truth label also, hence further investigation can be deployed to determine the exact label.

6.3.4 Forging Signature

Signature forgery refers to the act of falsely replicating the signature of another car. Adversaries familiar with forensic investigation can replicate the signature of a car to another car to misguide the forensic investigation. Several methods can be used to forge signatures. One of the easiest methods is to convolve the dashcam video of a car with blur kernels of another car.

Let us suppose Car A, and video from Car B are available. We aim to misguide the forensic investigation and make Car B video look like it is recorded from Car A. Towards this, we can record the video from Car A and extract blur kernel K_1. Since video from Car B is also available, we

Table 6.5: ***Similarity score S and ratio r computed using Eq. 6.3 and Eq. 6.4 respectively. '-' signifies ratio of 1. Numbers in bold represents maximum value of S and values of r less than $\tau = 2$ are marked in red. Possible predicted labels are computed by comparing value of r with τ, label of corresponding class is selected if ratio $r < \tau$. Similar results can be computed for other clips. (Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio)***

S.No	Video1		Video2		Video3		Video4	
GT / Ref.V	1		5		14		18	
	S	Ratio (r)	S	Ratio (r)	S	Ratio (r)	S	Ratio (r)
1	**9.615**	-	0.015	216.733	0.324	3.728	0.093	44.01
2	0.018	534.166	0.166	19.584	0.193	6.259	0.126	32.484
3	0.125	76.92	0.28	11.61	0.622	1.942	0.3	13.6433
4	0.041	234.512	0.117	27.786	0.026	46.461	0.304	13.463
5	0.016	600.9375	**3.251**	-	0.775	1.558	0.578	7.081
6	0.832	11.556	0.48	6.772	0.364	3.3186	0.144	28.423
7	0.01	961.5	0.764	4.255	0.375	3.2213	0.46	8.897
8	0.01	961.5	0.442	7.355	1.128	1.07	0.022	186.045
9	0.79	12.17	0.32	10.159	0.251	4.812	0.2644	15.48
10	0.237	40.569	0.044	73.886	0.028	43.142	2.2867	1.789
11	0.002	4807.5	0.948	3.429	0.551	2.192	0.0711	57.566
12	0.004	2403.75	0.124	26.217	0.788	1.532	0.051	80.254
13	0.347	27.708	0.467	6.961	0.528	2.287	0.226	18.11
14	0.416	23.112	0.806	4.033	**1.208**	-	0.0778	52.609
15	0.381	25.236	0.235	13.834	0.137	8.817	0.122	33.549
16	0.214	44.929	0.351	9.262	0.177	6.824	0.667	6.136
17	0.624	15.4086	0.495	6.567	0.968	1.247	0.1489	27.488
18	0.138	69.673	0.071	45.788	0.077	15.688	**4.093**	-
19	0.039	246.538	1.048	3.102	0.377	3.204	0.1378	29.702
20	0.084	114.464	0.662	4.91	0.331	3.649	0.556	7.361
21	0.055	174.818	0.996	3.264	0.106	11.396	0.1756	23.308
22	0.018	534.166	0.366	8.882	0.293	4.122	0.293	13.969
23	0.659	14.59	0.144	22.576	0.0178	67.865	0.267	15.329
24	0.035	274.714	0.217	14.981	0.64	1.887	0.2179	18.783
25	0.117	82.1794	0.44	7.388	0.2178	5.546	0.1178	34.745
τ	2		2		2		2	
Pred. Label(s)	1		5		3,5,8,12,14,17,25		10,18	

can extract blur kernel K_2 from video of Car B. We can perform forgery by applying following simple steps.

- Step 1: Extract K_1 and K_2 from respective videos.
- Step 2: De-convolve Car B video first using blur kernel K_2
- Step 3: Convolve Car B video with K_1.

A forensic investigation using the proposed algorithm will try to link the video under investigation to Car B. However since the video has been convolved with K_1, the proposed scheme for linking the video with the car will fail.

For testing the above hypothesis, we create a forged video databset using 20 videos from five different cars (Ecosport, Etios1, Grandi10, HondaCity and Jazz). We randomly select the blur signature from rest of the cars and

create the forged videos using above procedure. Videos are labeled using the random forest classifier as discussed in Section 6.1.4. We observe that all 20 videos are classified into the forged class. Table 6.6 shows results for 3 such videos. Video1 and Video2 belongs to class label 8 and forged with blur signature of class 14 and 1 respectively. However, after using RF classifier, labels of the forged classes are assigned to the videos.

Similarity test in case of forged videos

We further investigate such scenario by calculating similarity score of test videos with reference videos available to us. Motivation behind computing similarity score is that although forged video easily deceives RF classifier, it can still have some similarity with its original class. We perform the similarity test for all 20 videos, Table 6.6 shows the results for 3 videos. Using $\tau = 2$, we are able to predict multiple labels along with correct class for 15 videos. However, for other five videos even similarity test predicts forged label. In Table 6.6, we show best and worst case scenario. Predicted labels for Video1 are 8 and 14, since Video1 is forged with Class 14 signature, it is obvious to believe that it will have high similarity with Class 14, however even after forgery it has some similarity with Class 8. In case of Video2, it has no similarity with original class and the proposed algorithm fails to detect the similarity with other classes. It is interesting to note that in case of Video3 there are eight possible classes whose ratio is less than τ. However, if we observe carefully, possible labels also consists of original class. We can summarize that determining the exact class in case of forged video is difficult but using similarity measure, we can reduce the search space to few probable classes. Further investigation can be performed on reduced space to exactly determine the origin of video.

6.4 Application in Person Identification

Body-wear cameras are fast becoming standard kit for front line law enforcers and concerns have been raised over the handling of footage from police body wear cameras. These cameras are more than an extension of dashboard cameras for law enforcement. They are more versatile and more likely to present privacy issues [56], [59]. Recently many videos have surfaced over social media where perpetrator happens to record an act of his crime [51]. Linkage of video to a person is required to investigate cases where the perpetrator happens to be the video recorder using body wear camera.

Since the walk pattern of a person is known to be unique [131], [138], we hypothesize that the videos captured using body-wear cameras will carry these signatures. To evaluate our method under more principled settings, we collect a data set of 15 persons. The video recorder is attached to person's

Table 6.6: ***Predicting possible labels for forged videos using similarity score*** S ***and ratio*** r ***computed using Eq. 6.3 and Eq. 6.4.'-' signifies ratio of*** 1, ***corresponding maximum similarity scores are in bold and values of*** r ***less than*** $\tau = 2$ ***are marked in red. Possible predicted labels are computed by comparing value of*** r ***with*** τ, ***label of corresponding class is selected if ratio*** $r < \tau$. ***Similar results can be computed for other clips. (Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio)***

S.No	**Video1**		**Video2**		**Video3**	
GT	8		8		19	
Forged Class Label	14		1		11	
Predicted Label using Majority Voting	14		1		11	
Ref.V	**S**	**Ratio (r)**	**S**	**Ratio (r)**	**S**	**Ratio (r)**
1	0.1289	11.895	**4.7178**	-	0.0356	20.787
2	0.1289	11.895	0.0756	62.405	0.2178	3.398
3	0.1556	9.854	0.0822	57.394	0.2432	3.043
4	0.0867	17.685	0.0044	1072.227	0.0956	7.741
5	0.5622	2.727	0.36	13.105	0.7378	1.003
6	0.44	3.485	0.4089	11.538	0.2578	2.87
7	0.3044	5.037	0.0289	163.246	0.4067	1.82
8	0.9133	1.679	1.260	3.7443	0.6178	1.198
9	0.5111	3	0.1311	35.986	0.2711	2.73
10	0.1089	14.08	0.0511	92.325	0.0844	8.768
11	0.3489	4.395	0.1111	42.464	**0.74**	-
12	0.0622	24.651	0	Inf	0.4333	1.708
13	0.2178	7.04	0.2867	16.456	0.2889	2.561
14	**1.5333**	-	0.1867	25.269	0.7333	1.009
15	0.14	10.952	0.0489	96.479	0.08	9.25
16	0.2378	6.448	0.2644	17.843	0.1222	6.056
17	0.4889	3.136	0.3044	15.499	0.52	1.423
18	0.1933	7.932	0.0244	193.352	0.2778	2.664
19	0.2067	7.418	0.0422	111.796	0.4	1.85
20	0.4156	3.689	0.1378	34.237	0.4067	1.82
21	0.1556	9.854	0.2867	16.456	0.12	6.167
22	0.2978	5.149	0.0067	704.149	0.3222	2.297
23	0.0511	30.006	0.3711	12.713	0.0756	9.788
24	0.1978	7.752	0.1	47.178	0.0511	14.481
25	0.0756	20.282	0.0267	176.697	0.2	3.7
τ	2		2		2	
Pred. Label(s)	8,14		1		5,8,7,11,12,17,19,20	

Table 6.7: ***Table computes ratio r using similarity score S for video clips of 4 different persons. '-' signifies ratio of 1. Numbers in bold represents maximum similarity and values of r less than $\tau = 2$ are marked in red. Possible predicted labels are computed by comparing value of r with τ, label of corresponding class is selected if ratio $r < \tau$. (Ref.V = Reference Video, GT = Ground Truth Label, S = Similarity Score, r = ratio)***

S.no	Video1		Video2		Video3		Video4	
GT / Ref.V	1		4		8		9	
	S	r	S	r	S	r	S	r
1	**100.4121**	-	1.320	5.177	0.471	100.763	1.601	4.943
2	5.7584	17.43750	1.421	4.810	2.580	18.410	0.824	9.606
3	5.0386	19.92857	4.678	1.461	2.943	16.136	1.633	4.845
4	1.5979	62.8378	**6.838**	-	2.118	22.410	3.958	2.000
5	12.236	8.2058	2.076	3.292	1.061	44.745	0.762	10.377
6	1.709	58.736	2.350	2.909	2.184	21.746	1.979	4.000
7	0.608	165.088	1.785	3.830	9.213	5.1562	2.224	3.559
8	0.5830	172.222	2.540	2.691	**47.506**	-	2.746	2.883
9	1.328	75.609	3.253	2.101	2.418	19.642	**7.917**	-
10	0.816	122.907	2.127	3.214	2.825	16.815	4.318	1.833
11	1.533	65.492	0.996	6.859	0.982	48.351	0.773	10.232
12	5.974	16.8072	0.374	18.269	0.701	67.692	1.054	7.508
13	0.716	140.201	1.745	3.917	0.565	84.076	2.652	2.985
14	0.629	159.428	4.246	1.610	1.713	27.731	3.706	2.135
15	2.832	35.451	2.170	3.150	1.295	36.666	2.954	2.679
τ	2		2		2		2	
Pred.Label(s)	1		3,4,14		8		9,10	

chest. Each person recorded two 5 min sequence of video in morning and evening. We divide video into smaller clips similar to vehicle identification problem. Duration of each clip is approximately 30 seconds (1000 frames). We use same number of frames in each clip for simplicity. While creating clips for same person, there is difference of at least 200 frames between any two consecutive clips.

We now describe the process of creating training, testing and reference set. For each person 50% of clips (3000 frames) are selected randomly for creating training set. Reference and testing set for each class is created by selecting 25% of remaining clips in each set. Reference set is required for matching the test video with correct class/Person.

We calculate similarity for each clip in testing set with all the clips present in reference set using Eq. 6.3. Therefore, for any particular clip we will have 15 similarity scores, one for each Person. We decide if the clip is recorded by the particular person if the similarity score with corresponding reference clip is maximum. Fig 6.10 shows the similarity score of 15 test clips with

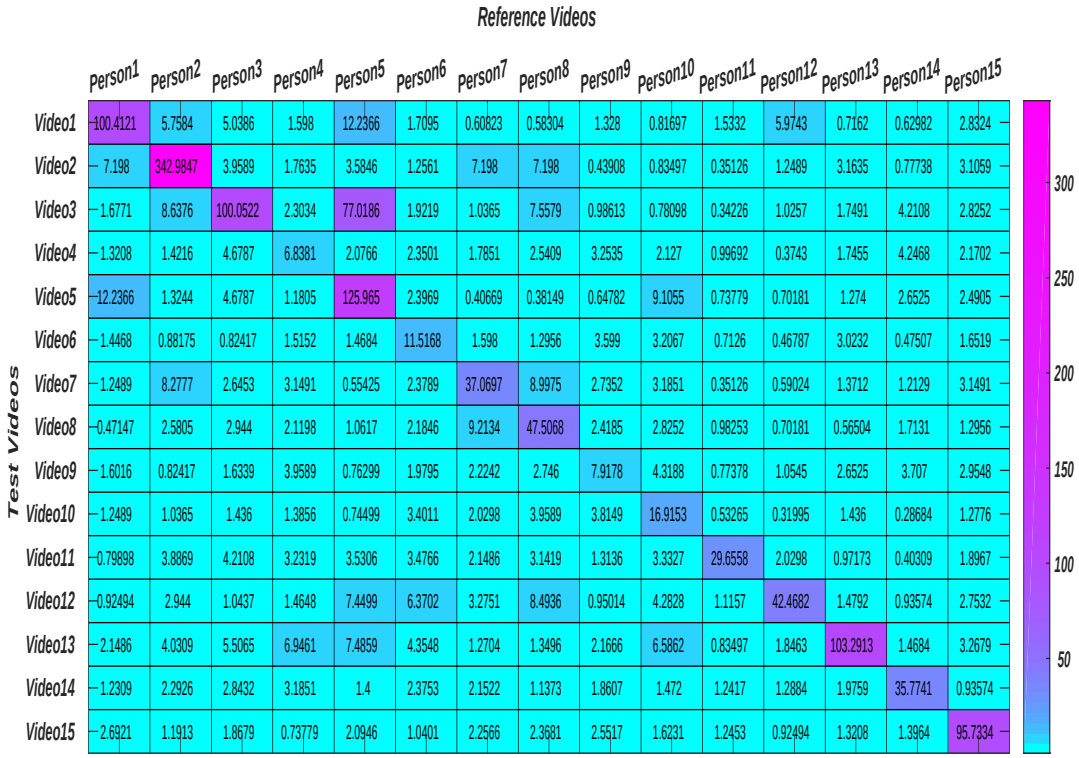


Figure 6.10: ***Similarity score for 15 different people computed using Eq. 6.3. Test and reference clips have same number of frames (1000 frames each). Higher value represents higher similarity between reference clip and test clip. Each cell represents similarity score of a video (in vertical axis) with reference video of corresponding person (in horizontal axis).***

reference set[3]. We can observe from Fig 6.10 that similarity score for any particular clip is high if it belongs to same person. Otherwise, similarity score is significantly low. For example, in case test clip belongs to person 9, maximum similarity score is 7.9178 which is at least two times higher than next maximum value (4.3188 for Person 10). But in some cases it may happen that the second closest score may be very near to the first. Therefore, to further improve our confidence, we compute ratio r using Eq. 6.4 and follow similar procedure used in case of vehicles. We decide whether test clip is similar to reference clip if value of r for that particular class is less than predetermined τ. In Table 6.7, we show r for best and the worst performing cases. Hence, procedure described above can reduce the search space from very large number of classes to few probable classes which is required in most of the forensics scenarios. For example, in case of Clip2 and Clip4 (Table 6.7), the search space is reduced to classes 3, 4, 14 in case of former and 9, 10 for latter. Similar results can be computed for other clips.

For calculating similarity using random forest, we compute the number of feature vectors terminating at same node position in each decision tree and plot their histogram. Fig 6.11 shows comparison between histogram plots of

[3]To make the figure readable, we arrange the test videos according to Person number. It is not necessary that they are tested in same sequence

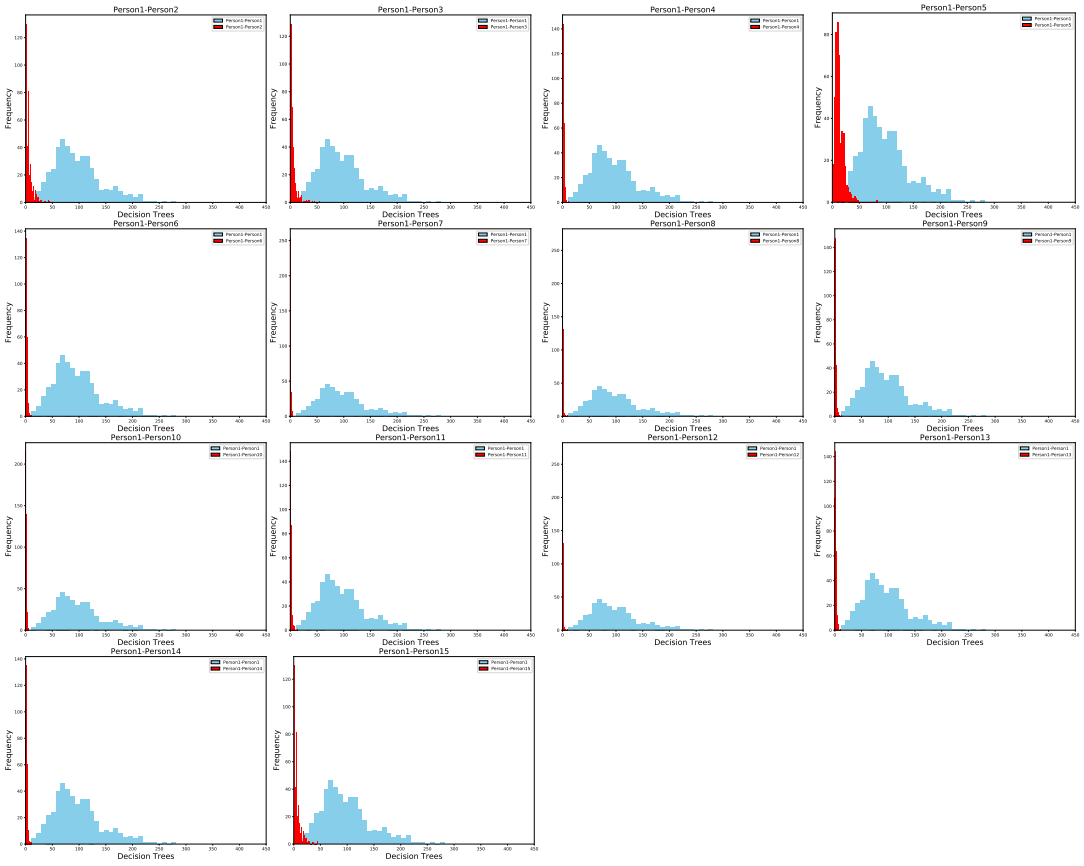


Figure 6.11: ***Histogram plot of the number of feature vectors terminating at the same node position in each decision tree. Blue histogram represent test and reference video belongs to same person, whereas the histogram in red represents every other case. Figure provides comparison of test video of Person1 with reference videos of every other person in the dataset. We can observe that when clip belongs to same person, many decision trees have feature vectors terminating at same node position, however in other case only few decision trees have feature vectors terminating at same node position. Such information can be used to differentiate between two clips, since the distribution is different for the two cases. (Please refer to the web version for better visualization. Decision trees on x-axis and Frequency on y-axis)***

clip recorded by Person 1 compared with reference clips of every other person. Blue histogram represents test clip of Person 1 compared with reference clip of Person 1, whereas histogram in red represents all other cases. We can observe that when clip belongs to the same person, many decision trees have feature vectors termination at same node position, however in other case only few decision trees have features vectors at same node position. We can also use this information for determining the similarity between the clips since there is clear difference in distribution in both of the cases.

6.5 Discussion

The camera mounted inside vehicle undergoes the same motion as the vehicle does and also experiences the vibrations produced due to engine. Each vehicle can have unique vibrations and motion pattern. Therefore, the blur generated in videos captured by dashcams have a unique signature which can be used for classifying the vehicle. The use of the blur pattern produced due to the motion of a camera for classifying the vehicle is a novel approach and can be extended to other categories of vehicles too.

In order to show the effectiveness of our approach, we perform elaborate experiments. Firstly, it is important to show that the blur pattern generated from a video from a particular vehicle is unique. We address this concern by creating a database which consists of videos recorded during different days, location, time and even driven by different drivers. It is also evident from experiments that blur pattern is unique even if extracted from videos from different days, location and time.

Resolution of a video is also an important factor while estimating blur. It is observed from experiments that classes with low-resolution videos perform poorer than classes with higher resolution videos. Hence, for experiments, we only use the videos with resolution greater than or equal to 480p.

To prove that unique blur pattern is due to motion and vibrations produced due to an engine, we use different cameras to record videos. We can see from experiments that our algorithm can identify the vehicle from such videos with considerable accuracy. This shows that blur pattern is generated due to vibration of engine and motion of a vehicle which is unique to the particular vehicle.

7 Conclusion and Future Directions

7.1 Conclusion

In this work, we estimate the robust signature from multimedia and apply them for forensic analysis tasks such as camera identification, forgery detection and localization, vehicle forensics. We also propose a framework for concealing the traces of contrast enhancement and suppressing the signatures in dashcam videos.

The main limitation of source camera identification is that due to in-camera transformation, estimation of PRNU is also affected, which leads to erroneous source identification. We address this problem by estimating PRNU from probabilistically obtained raw data. We propose the model for probabilistically obtained raw measurements and develop the theory for estimating PRNU from the same. We show that the obtained estimate is MVU and determine bound on its variance using CRLB. We make use of PCE as decision statistics due to its property of reducing the false positives in comparison with other decision statistics. We compare our results with the other two popular estimation algorithms, MLE-PRNU and Phase-PRNU and show that there is a considerable improvement in classification accuracy. We test our proposed approach for patch size varying from 512×512 to 64×64. The performance gain of the proposed algorithm can also be observed from tamper localization results. From observed results, we can conclude that localization of forgery can be improved by using the proposed algorithm for estimation of PRNU.

In our second objective, we propose a joint optimization framework for adaptive contrast enhancement anti-forensics. The proposed algorithm uses information from the spatial and DCT domain for generating the contrast-enhanced images while suppressing the typical CE artifacts. We present an analytical approach to optimize the objective function. It can be observed from the experimental results that enhanced images generated using the proposed scheme are robust to the wide variety of CE detectors. Further, the proposed algorithm also performs better than other counter forensic approaches in terms of image quality assessment metrics.

In Chapter 6, we propose a novel method for vehicle forensics using multimedia signatures. We emphasize and discuss the crucial need for vehicle forensics for security and privacy issues caused by the remarkable increase

in the use of dashcams. Currently, there are no algorithms that deal with egocentric video analysis for vehicles. Our algorithm extracts motion blur from dashcam videos, which can act as a unique signature for classifying the vehicle correctly. We show with the help of experiments that the blur pattern is unique to a particular vehicle and is only dependent on its motion and vibratory signature. There are no effects of changing the camera and the location on blur patterns. We also investigate the possibility of forged videos with alien blur kernels and propose a countermeasure.

Extending our proposed approach to first-person identification using a chest-mounted camera, we demonstrate that the scope of the problem is not only limited to cameras mounted inside the car. There are many open research problems associated with this work. One limitation of the proposed approach is to study the effect of enlarging the dataset or increasing the number of cars of similar models. In the future, we would work towards enhancing the dataset and issues related to the intersection of driving behavior and security concerns.

7.2 Future Directions

The objective of this research is to develop robust algorithms for estimation and concealment of forensic multimedia signatures. Also, with the introduction of a new research area of egocentric analysis of dashcam videos, several challenging problems are highlighted which require further investigation. We have divided the future work into two categories. In the first category, we discuss the problems that can be extended from the work presented in this thesis, and in the second category, we discuss the new problems with a broader view on the entire image forensic field.

- There are several lines of research arising from this work that can be pursued.
  - **Deep learning-based forensic detectors**: Currently, due to the success of deep learning in many multimedia processing tasks, there is a high interest in CNN-based detectors. Recent research shows that camera identification and tampering detection task can be achieved using CNN [122], [100], [157], [179], [6], [20]. Using deep learning for multimedia forensics is an active research area with a large scope of improvement in preliminary results. In one such problem, locating the altered regions in a tampered picture is a challenging task because the difference is unknown between the altered regions and the original regions. Therefore, for exposing a certain type of tampering, an assumption is first proposed, and then an algorithm is designed under such an assumption. Hence for convincing results, it is necessary to search

large hypothesis space [122]. Deep networks can be used for solving this difficult problem as a neural network can be trained over the large dataset and can give state-of-the-art results.

– **Tampering detection using region proposals**: Region proposals are widely used in object detection and segmentation of images. With technological advances leading to an increase in mechanisms for image tampering, tamper localization algorithms should also follow a similar path.

  One problem with current algorithms is that they require prior knowledge of the method of forgery in order to localize the forgery. Also, while using machine learning algorithms to learn different types of tampering, we are still left with the question of which features to train on, and how to localize the manipulation. To solve this, object detection networks such as Faster R-CNN, which can combine a region proposal network with a CNN, have recently been adapted to fraud detection by utilizing their ability to propose the bounding box for objects of interest to localize the tampering artifacts [188], [182]. Since most forgery is performed form insertion or removal of objects in a given scene, computing region proposal and then computing normalized statistics on them can lead to better results.

– **Adversarial Multimedia forensics**: The disadvantage of the proposed approach is that we assume the analyst is not aware of the attempts and counter forensic techniques adopted by the forger. The counter forensic tools are developed to impede the performance of forensics detectors. In general, they are designed to attack a specific algorithm without paying attention to the possible countermeasure adopted by the analyst, e.g., by neglecting the fact that the CF attack may itself leave traces that can be revealed by the analyst. On the other hand, anti counter forensic techniques are developed, often by targeting specific counter forensics techniques without taking into account the possibility that the attacker foresees the move of the analyst. In such a scenario, forger and analyst will keep developing algorithms to counter each other independently in a never-ending loop. Therefore, for understanding the real security issues, it becomes necessary to develop the framework in a game-theoretic view [14]. The new approaches based on adversarial multimedia forensics can be induced from techniques described in the literature, such as [12], [10], [15]. The game-theoretic view in multimedia forensics is a newly evolving area. The work proposed in this thesis can be extended by incorporating the game theory framework for addressing the security requirements in real-world scenarios.

– **GAN based image enhancement detection algorithms**: The research in adversarial networks shows that GAN's can also be used

to improve the robustness and enhance the statistical difference between original images and filtered or enhanced images. For example, authors in [101], proposed the framework of generative adversarial networks to generate images that follow the underlying statistics of unaltered images, thereby significantly enhancing forensic undetectability. Therefore in the future, there is a need for building GAN based image enhancement detection algorithms.

- **Studying the driving behavior**: Using research presented in *Chapter 6: Egocentric analysis of Dashcam videos* as a baseline, several other challenging open problems can be investigated. One such problem is studying the driving behavior of the individual driver. Analysis of videos from dashcams can provide valuable feedback to the driver about his/her driving behavior [136]. Such information can be invaluable since it can help in reducing the occurrence of accidents and thus save lives. Also, videos generated by egocentric cameras, whether it is worn by the person, motorist, or mounted inside the car, have security and privacy issues associated with them. Since the use of egocentric cameras is increasing rapidly, there are a lot of long term consequences that we probably won't see for a while. However, we are already starting to see the importance of several key issues like privacy protection for civilians recorded by the cameras. There are many open research problems associated with this work.
- **Egocentric dashcam video analysis using deep learning models**: As future work, transfer learning models can be explored. Siamese architectures are also a good way of comparing the samples.

- Besides the problems discussed above, the following new problems can also be explored
  - **PRNU based image clustering with deep learning**: Practical applications of digital forensics are often faced with the challenge of grouping large-scale suspicious images into a vast number of clusters, each containing images taken by the same camera [134]. This task can be approached by resorting to the use of sensor pattern noise (SPN), which serves as the fingerprint of the camera. The challenges of large-scale image clustering come from the sheer volume of the image set and the high dimensionality of each image.

    Fig 7.1 shows a plausible pipeline for image clustering. Resnet50 provides with the ability to extract features automatically from high pass filtered image [37]. The output of a fully connected layer of Resnet50 can be used as PRNU features, whereas the softmax layer can provide us with the number of clusters. The number of cluster centers is updated by updating the classification output of the Resnet50 network.

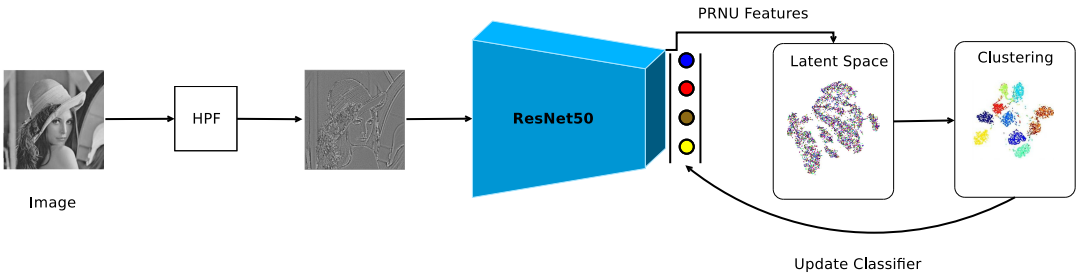


Figure 7.1: ***The pipeline for PRNU based image clustering with deep learning.***

– **Detecting deep fakes videos**: Deep fake is an AI-based technique used to produce or alter video content [38], [70], [73], [74], [97], [192]. Plenty of research is available on image and video forensics. However, detecting AI-generated content is a relatively new area [78], [132]. Some forensic techniques try to detect face swap videos by observing blink or no blink at the expected frequency [117]. This is possible because the data used to synthesize faces typically did not involve the person with their eyes close. However, by including blinking in training dataset makes this technique less effective. In another similar work, forensic technique models facial expressions and movements for representing an individual's speaking pattern [1]. Although not visually apparent, these patterns are often violated by the nature of how deep-fake videos are created and can, therefore, be used for authentication.

  Discovering subtle cues in GAN-manipulated images and videos can allow us to detect the GAN generated images. GAN's can mimic the data distribution, however, producing the same distribution of noise pattern as generated by camera pipeline in original images and videos is required to be investigated. Identifying the features which are only present in images produced by the camera, we can differentiate between GAN and camera-generated images. Developing such tools is an area of active research and can significantly impact our society.

8 Author's Publications

8.1 Journal

- **A. Mehrish**, A. V. Subramanyam and S. Emmanuel, "Sensor Pattern Noise Estimation Using Probabilistically Estimated RAW Values," in IEEE Signal Processing Letters, vol. 23, no. 5, pp. 693-697, May 2016.
- **A. Mehrish**, A.V. Subramanyam, S, Emmanuel, "Robust PRNU estimation from probabilistic raw measurements", Signal Processing: Image Communication, Volume 66, 2018, Pages 30-41, ISSN 0923-5965
- **A. Mehrish**, A. V. Subramanyam and S. Emmanuel, "Joint Spatial and Discrete Cosine Transform Domain based Counter Forensics for Adaptive Contrast Enhancement," in IEEE Access. doi: 10.1109/ACCESS.2019.2901345
- **A. Mehrish**, P. Singh, P. Jain, A. V. Subramanyam and M. Kankanhalli, "Egocentric Analysis of Dash-cam Videos For Vehicle Forensics" in IEEE Transactions on Circuits and Systems for Video Technology. doi: 10.1109/TCSVT.2019.2929561.

8.2 Conference

- S. Sharma, A. V. Subramanyam, M. Jain, **A. Mehrish** and S. Emmanuel, "Anti-forensic technique for median filtering using L1-L2TV model," 2016 IEEE International Workshop on Information Forensics and Security (WIFS), Abu Dhabi, 2016, pp. 1-6.
- **A. Mehrish**, A. V. Subramanyam and M. Kankanhalli, "Multimedia signatures for vehicle forensics," 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, 2017, pp. 685-690.

A RAW to MATLAB

In this appendix, we present a step-by-step guide on how to read a RAW image into MATLAB and process the raw sensor information into a correctly usable image[1].

- First, run *dcraw* in verbose mode with 'as shot' white balance to run it through the entire processing chain and output some useful information. Once dcraw has been compiled and is in your path, you can simply get this information (and a preliminary image) from the terminal command line by typing

  ```
  $ dcraw -v -w -T <raw_file_name>
  ```

- To get the raw sensor data into MATLAB, first we use dcraw with the following options to output a 16bpp TIFF file. This will also overwrite the previously produced 'recon image.'

  - -4 : writes linear 16-bit, unbrightened and un-gamma-corrected image, same as '-6 -W -g 1 1'
  - -D : Foregoes demosaicing
  - -T : Writes to TIFF file instead of PPM

  Thus we output the simple sensor data from the command line with the following:

  ```
  $ dcraw -4 -D -T <raw_file_name>
  ```

- You can now read this file into MATLAB using

  ```
  $ raw = double(imread('file.tiff')}
  ```

  which will yield the raw CFA information of the camera. The image will be slightly larger than the pixel dimensions quoted by the camera, but that is because dcraw does not discard some of the valid border pixels that most programs do.

[1] Further details can be found in https://rcsumner.net/raw_guide/RAWguide.pdf

B Blur Extraction

We briefly discuss the algorithms used for computing blur kernel from Eq. 6.2. According to [105], problem can be divided into following sub problem

Y sub-problem

$$\min_{Y} \lambda \|KY - Q\|_2^2 + \frac{\|Y\|_1}{\|Y\|_2} \tag{B.1}$$

K sub-problem

$$\min_{K} \lambda \|KY - Q\|_2^2 + \varphi \|K\|_1 \tag{B.2}$$

The regularizer in Equation B.1 makes it non-convex. If the denominator of regularizer from previous iteration is considered to be a constant it can be converted into l_1-regularized problem which is convex and solvable. Hence Equation B.1 can also be written as

$$\min_{Y} \|KY - Q\|_2^2 + \mu \|Y\|_1 \tag{B.3}$$

where K is the kernel matrix.

The problem described in Equation B.3 is l_1-regularized least squares. It is an unconstrained optimization problem with a non-differentiable cost function. There are many algorithms in literature for obtaining the optimal solution to Equation B.3. We use the Bregman Iterative Algorithm (BIA) [190] to obtain the solution for the above equation. BIA is used because for the moderate value of the penalty parameter μ, accurate solution to original problem in Equation 6.1 is obtained with small cost of solving a single instance of Equation 6.2.

In Algorithm 2, $D_J^{p^k}$ is the Bregman Distance [190] and is defined as:

$$D_J^{p^k}(u, v) = J(u) - J(v) - < p^k, u - v > \tag{B.4}$$

The inner iteration of the Y sub-problem algorithm is solved using BIA while the outer loop simply re-estimates the weighting on the likelihood term in Equation 6.2 by updating the denominator $\|Y\|_2$. The overall procedure is

given in Algorithm 1. Kernel estimate K is updated after the estimate of Y is obtained. The K sub-problem is given by

$$\min_{K} \lambda \|KY - Q\|_2^2 + \varphi \|K\|_1 \tag{B.5}$$

subject to the constraints $K \geq 0$, $\Sigma_i K_i = 1$.

Algorithm 1 Blur Extraction Algorithm

Input F, Max. kernel size h
Output K

procedure BLURExtraction
  $Q \leftarrow [\nabla_x F, \nabla_y F]$
  Initialize $Y^0 \leftarrow 0, K^0 \leftarrow 0, i \leftarrow 0$
  Loop over coarse-to-fine levels
    Alternate:
    $Y^{i+1} \leftarrow updateframe(Y^i, K^i, Q)$
    $K^{i+1} \leftarrow updatekernel(Y^i, K^i, Q)$
    $i \leftarrow i + 1$
    Interpolate solution to finer level as initialization
  return blur matrix K
end procedure

Input Blur kernel K from previous K sub-problem solution, estimated sharp frame Y from previous Y sub-problem, observed blurry frame Q.

procedure updateFrame
  Regularization parameter $\mu = 0.05$
  Max outer iterations m = 2, Inner iterations n = 2
  for j = 0 to $m - 1$ **do**
    $\grave{u} = \mu \|Y^j\|_2$
    $Y^{j+1} = BIA(\grave{u}, Y^j, Q, K, n)$
  end for
  return: updated frame Y^m
end procedure

Input Blur kernel K from previous K sub-problem solution, estimated sharp frame Y from previous Y sub-problem, observed blurry frame Q.

procedure updateKernel
  Solve Eq. B.2 using IRLS.
  return: updated kernel K
end procedure

Unconstrained iterative re-weighted least squares (IRLS) [111] is used for solving above equation followed by setting negative elements of estimated K to 0, and re-normalizing. IRLS is executed for 1 iteration with weights being computed from the kernel of the previous K sub-problem solution. The conjugate gradient (CG) is used to solve the inner IRLS system. Once kernel is estimated, thresholding is done for increasing the robustness to

Algorithm 2 Bregman Iteration Algorithm

Input: μ, Y, Q, K, n
Initialize $Y^0 \leftarrow 0$, $p^0 \leftarrow 0$
procedure BIA
    $J(Y) = \mu\|Y\|_1$
    $H(Y) = \|KY - Q\|_2^2$
    for k =0,.., $n-1$ **do**
        $Y^{k+1} \leftarrow \arg\min_Y D_J^{p^k}(Y, Y^k) + \|KY - Q\|^2$
        $p^{k+1} \leftarrow p^k - 2K^T(KY^{k+1} - Q)$
    end for
    Return: Y^n
end procedure

noise [68], [185] by making small elements of the kernel equal to zero. We successfully estimated the spatially variant blur using this algorithm.

C Parameter Estimation: Generalized Gaussian Distribution

C.1 Mathematical Model

The problem is to estimate the parameters of Generalized Gaussian Distribution (GGD) using maximum likelihood estimation (MLE) method. The probability density function of GGD is given by

$$f(\Delta;\alpha,\beta,\mu) = \frac{\beta}{2\beta\Gamma(\frac{1}{\beta})}\exp\left(-\frac{|\Delta-\mu|^{\beta}}{\alpha}\right) \tag{C.1}$$

where $\Gamma(\bullet)$ is the Gamma function given by $\Gamma(x) = \int_0^{\infty}\exp^{-t}t^{x-1}dt$ $(x>0)$ μ, α, $\beta > 0$ denotes mean, scale and shape parameter respectively.

C.2 Maximum Likelihood Estimation

We assume N points x_i be given in interval $[a,b]$. Each x_i is distributed according to C.1.

The likelihood function is given by

$$\begin{aligned}\mathcal{L} =& \prod_{i=1}^{N} f(x_i) = \Big(\frac{\beta}{2\alpha\Gamma(\frac{1}{\beta})}\Big)^{N}\exp\Big(-\sum_{i=1}^{N}\Big(\frac{|x_i-\mu|}{\alpha}\Big)^{\beta}\Big)\\ =&(2\alpha)^{-N}\beta^{N}(\Gamma(\frac{1}{\beta}))\exp\Big(-\sum_{i=1}^{N}\Big(\frac{|x_i-\mu|}{\alpha}\Big)^{\beta}\Big)\end{aligned} \tag{C.2}$$

The log-likelihood of a $\mathcal{L}$ is given by

$$L = \log(\mathcal{L}) = -N\log(2\alpha) + N\log(\beta) - N\log(\Gamma(\frac{1}{\beta})) - \sum_{i=1}^{N}\Big(\frac{|x_i-\mu|}{\alpha}\Big)^{\beta} \tag{C.3}$$

In this derivation we assume that μ is estimated using sample mean

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{C.4}$$

For obtaining maximum likelihood estimate of α and β, differentiate L with respect to α and β and set their derivative to zero.

$$\frac{\partial L}{\partial \alpha} = -\frac{N}{2\alpha} - \beta \sum_{i=1}^{N} \frac{(|x_i - \mu|)^{\beta}}{\alpha^{\beta+1}} \tag{C.5}$$

$$\frac{\partial L}{\partial \beta} = \frac{1}{\beta} + \frac{\Psi(\frac{1}{\beta})}{\beta^2} - \frac{1}{N}\sum_{i=1}^{N} \left|\frac{x_i - \mu}{\alpha}\right|^{\beta} \log\left|\frac{x_i - \mu}{\alpha}\right| \tag{C.6}$$

Setting C.5 to 0, estimate of α can be obtained as

$$\alpha = \left(\frac{\beta}{N}\sum_{i=1}^{N} |x_i - \mu|^{\beta}\right)^{\frac{1}{\beta}} \tag{C.7}$$

Solving for $\frac{\partial L}{\partial \beta} = 0$ is not straightforward. To get the estimate of β main idea is to substitute the value of α computed in C.5 in C.6. Then after multiplying with the factor $\frac{\beta}{N}$, we get

$$g(\beta) = 0 \tag{C.8}$$

where $g(\beta)$ is given by

$$g(\beta) = 1 + \frac{\Psi(\frac{1}{\beta})}{\beta} - \frac{\sum_{i=1}^{N} |x_i - \mu|^{\beta} \log|x_i - \mu|}{\sum_{i=1}^{N} |x_i - \mu|^{\beta}} + \frac{\log(\frac{\beta}{N}\sum_{i=1}^{N} |x_i - \mu|^{\beta})}{\beta} \tag{C.9}$$

where Ψ is digamma function

$$\Psi(x) = \frac{\partial \log(\Gamma(x))}{\partial x} = \frac{\Gamma'(x)}{\Gamma(x)} \tag{C.10}$$

β is estimated from C.9 by using a Newton-Raphson iterative procedure, starting from an initial guess of $\beta = \beta_0$, where β_0 is given by

$$\beta_0 = \frac{m1}{\sqrt{m2}} \tag{C.11}$$

where $m1$ is the first statistical moment of the absolute values and m_2 is the second statistical moment. The iteration is give by

$$\beta_{i+1} = \beta_i - \frac{g(\beta_i)}{g^{'}(\beta_i)} \tag{C.12}$$

where $g(\beta)$ is given by equation C.9 and $g^{'}(\beta)$ is the derivative of $g(\beta)$.

$$\begin{aligned} g^{'}(\beta) = &- \frac{\Psi(\frac{1}{\beta})}{\beta^2} - \frac{\Psi^{'}(\frac{1}{\beta})}{\beta^3} + \frac{1}{\beta^2} - \frac{\sum_{i=1}^{N} |x_i - \mu|^{\beta} (\log |x_i - \mu|^2)}{\sum_{i=1}^{N} |x_i - \mu|^{\beta}} \\ &+ \Big(\frac{\sum_{i=1}^{N} |x_i - \mu|^{\beta} \log |x_i - \mu|}{\sum_{i=1}^{N} |x_i - \mu|^{\beta}} \Big)^2 + \frac{\sum_{i=1}^{N} |x_i - \mu|^{\beta} \log |x_i - \mu|}{\beta \sum_{i=1}^{N} |x_i - \mu|^{\beta}} \\ &- \frac{log(\frac{\beta}{N} \sum_{i=1}^{N} |x_i - \mu|^{\beta})}{\beta^2} \end{aligned} \tag{C.13}$$

where $\Psi^{'}$ is a trigamma function

$$\Psi^{'}(x) = \frac{\partial^2 \log(\Gamma(x))}{\partial x^2} \tag{C.14}$$

Bibliography

[1] Darius Afchar et al. "Mesonet: a compact facial video forgery detection network". In: *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE. 2018, pp. 1–7.

[2] Solange Akselrod, Jacob Karin, and Michael Hirsch. "Fetal ECG Detection and Applications". In: *Advances in Processing and Pattern Analysis of Biological Signals*. Ed. by Isak Gath and Gideon F. Inbar. Boston, MA: Springer US, 1996, pp. 297–305. ISBN: 978-1-4757-9098-6. DOI: 10.1007/978-1-4757-9098-6_20. URL: https://doi.org/10.1007/978-1-4757-9098-6_20.

[3] K.R. Akshatha et al. "Digital camera identification using PRNU: A feature based approach". In: *Digital Investigation* 19.Supplement C (2016), pp. 69 –77. ISSN: 1742-2876. DOI: https://doi.org/10.1016/j.diin.2016.10.002.

[4] Sakinah Ali Pitchay and Ata Kabán. "Estimation of the Regularisation Parameter in Huber-MRF for Image Resolution Enhancement". In: *Intelligent Data Engineering and Automated Learning – IDEAL 2013*. Ed. by Hujun Yin et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 294–301.

[5] E. J. Alles, Z. J. M. H. Geradts, and C. J. Veenma. "Source Camera Identification for Heavily JPEG Compressed Low Resolution Still Images". In: *Journal of Forensic Sciences* 54 (2009), pp. 628–638.

[6] I. Amerini et al. "Localization of JPEG Double Compression Through Multi-domain Convolutional Neural Networks". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2017, pp. 1865–1871. DOI: 10.1109/CVPRW.2017.233.

[7] Irene Amerini et al. "Dealing with video source identification in social networks". In: *Signal Processing: Image Communication* 57.Supplement C (2017), pp. 1 –7. ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2017.04.009.

[8] Yaniv Bar, Noga Levy, and Lior Wolf. "Classification of Artistic Styles Using Binarized Features Derived from a Deep Neural Network". In: *Proc. Springer ECCV*. Ed. by Lourdes Agapito, Michael M. Bronstein, and Carsten Rother. 2015.

[9] M. Barni, E. Nowroozi, and B. Tondi. "Higher-order, adversary-aware, double JPEG-detection via selected training on attacked samples". In: *2017 25th European Signal Processing Conference (EUSIPCO)*. 2017, pp. 281–285. DOI: 10.23919/EUSIPCO.2017.8081213.

[10] Mauro Barni, Zhipeng Chen, and Benedetta Tondi. "Adversary-aware, data-driven detection of double JPEG compression: How to make

counter-forensics harder". In: *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*. IEEE. 2016, pp. 1–6.

[11] Mauro Barni, Marco Fontani, and Benedetta Tondi. "A universal technique to hide traces of histogram-based image manipulations". In: *Proc. of the on Multimedia and security*. ACM. 2012, pp. 97–104.

[12] Mauro Barni, Ehsan Nowroozi, and Benedetta Tondi. "Higher-order, adversary-aware, double JPEG-detection via selected training on attacked samples". In: *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 281–285.

[13] Mauro Barni and Fernando Pérez-González. "Coping with the enemy: Advances in adversary-aware signal processing". In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE. 2013, pp. 8682–8686.

[14] Mauro Barni, Matthew C Stamm, and Benedetta Tondi. "Adversarial multimedia forensics: Overview and challenges ahead". In: *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE. 2018, pp. 962–966.

[15] Mauro Barni et al. "Cnn-based detection of generic contrast adjustment with jpeg post-processing". In: *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE. 2018, pp. 3803–3807.

[16] S. Bayram et al. "Source camera identification based on CFA interpolation". In: *IEEE International Conference on Image Processing 2005*. Vol. 3. 2005, pp. III–69–72. DOI: 10.1109/ICIP.2005.1530330.

[17] Sevinç Bayram, Husrev T Sencar, and Nasir D Memon. "Seam-carving based anonymization against image & video source attribution". In: *2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2013, pp. 272–277.

[18] Sai Bi, Xiaoguang Han, and Yizhou Yu. "An L 1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition". In: *ACM Transactions on Graphics (TOG)* 34.4 (2015), p. 78.

[19] Rainer Böhme and Matthias Kirchner. "Counter-forensics: Attacking image forensics". In: *Digital image forensics*. Springer, 2013, pp. 327–366.

[20] L. Bondi et al. "Tampering Detection and Localization Through Clustering of Camera-Based CNN Features". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2017, pp. 1855–1864.

[21] Thierry Bouwmans et al. *Background Modeling and Foreground Detection for Video Surveillance*. CRC Press, 2014.

[22] Leo Breiman. "Random Forests". In: *Machine Learning* 45.1 (2001), pp. 5–32.

[23] A. Buades, B. Coll, and J. . Morel. "A non-local algorithm for image denoising". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 2. 2005, 60–65 vol. 2. DOI: 10.1109/CVPR.2005.38.

[24] R. Caldelli et al. "Fast image clustering of unknown source images". In: *2010 IEEE International Workshop on Information Forensics and Security*. 2010, pp. 1–5.

[25] Gang Cao, Yao Zhao, and Rongrong Ni. "Forensic estimation of gamma correction in digital images". In: *Proc. of IEEE International Conference on Image Processing*. 2010, pp. 2097–2100.

[26] Gang Cao et al. "Anti-forensics of contrast enhancement in digital images". In: *Proc. of the 12th ACM Workshop on Multimedia and Security*. 2010, pp. 25–34.

[27] Gang Cao et al. "Attacking contrast enhancement forensics in digital images". In: *Science China Information Sciences* 57 (2014), pp. 1–13.

[28] Gang Cao et al. "Contrast enhancement-based forensics in digital images". In: *IEEE transactions on information forensics and security* 9.3 (2014), pp. 515–525.

[29] X. Cao et al. "Visual Attention Accelerated Vehicle Detection in Low-Altitude Airborne Video of Urban Environment". In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.3 (2012), pp. 366–378. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2011.2163443.

[30] O. Celiktutan, B. Sankur, and I. Avcibas. "Blind Identification of Source Cell-Phone Model". In: *IEEE Transactions on Information Forensics and Security* 3.3 (Sept. 2008), pp. 553–566.

[31] Miriam Cha, Youngjune L. Gwon, and H.T. Kung. "Geolocation with Subsampled Microblog Social Media". In: *Proceedings of the 23rd ACM International Conference on Multimedia*. MM '15. Brisbane, Australia: ACM, 2015, pp. 891–894. ISBN: 978-1-4503-3459-4. DOI: 10.1145/2733373.2806357. URL: http://doi.acm.org/10.1145/2733373.2806357.

[32] A. Chakrabarti et al. "Modeling Radiometric Uncertainty for Vision with Tone-mapped Color Images". In: *Image Processing, IEEE Transactions on.* 36 (2014), pp. 2185–2198.

[33] Kyong I Chang, Kevin W Bowyer, and Patrick J Flynn. "An evaluation of multimodal 2D+ 3D face biometrics". In: *IEEE transactions on pattern analysis and machine intelligence* 27.4 (2005), pp. 619–624.

[34] M. Chen and J. Fridrich. "Determining image origin and integrity using sensor noise". In: *Information Forensics and Security, IEEE Transaction on* 3 (Mar. 2008), pp. 74–90.

[35] M. Chen et al. "Source digital camcorder identification using sensor photo response non-uniformity". In: *Proceedings of SPICE*. Vol. 6505. 2007, 65051G.

[36] S. Chen et al. "Automatic Detection of Object-Based Forgery in Advanced Video". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.11 (2016), pp. 2138–2151. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2015.2473436.

[37] Yunshu Chen, Yue Huang, and Xinghao Ding. "Camera model identification with residual neural network". In: *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2017, pp. 4337–4341.

[38] Robert Chesney and Danielle Citron. "Deepfakes and the New Disinformation War: The Coming Age of Post-Truth Geopolitics". In: *Foreign Aff.* 98 (2019), p. 147.

[39] G. Chierchia et al. "A Bayesian-MRF Approach for PRNU-Based Image Forgery Detection". In: *IEEE Transactions on Information Forensics and Security* 9.4 (2014), pp. 554–567.

[40] G. Chierchia et al. "Guided filtering for PRNU-based localization of small-size image forgeries". In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2014, pp. 6231–6235.

[41] Pedro Comesana-Alfaro and Fernando Pérez-González. "Optimal counterforensics for histogram-based forensics". In: *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*. 2013, pp. 3048–3052.

[42] A. Cortiana et al. "Performance comparison of denoising filters for source camera identification". In: *Proceedings of SPIE 7880, Media Watermarking, Security, and Forensics III* (Feb. 2011).

[43] Filipe de O Costa et al. "Open set source camera attribution and device linking". In: *Pattern Recognition Letters* 39 (2014), pp. 92–101.

[44] Davide Cozzolino, Diego Gragnaniello, and Luisa Verdoliva. "A novel framework for image forgery localization". In: *CoRR* abs/1311.6932 (2013). URL: http://arxiv.org/abs/1311.6932.

[45] K. Dabov, V. Katkovnik, and K. Egiazian. "Image denoising by sparse 3D transform-domain collaborative filtering". In: *Image Processing, IEEE Transactions on.* 16 (2007), pp. 2080–2095.

[46] Kostadin Dabov et al. "Image restoration by sparse 3D transform-domain collaborative filtering". In: *Image Processing: Algorithms and Systems VI*. Ed. by Jaakko T. Astola, Karen O. Egiazarian, and Edward R. Dougherty. Vol. 6812. International Society for Optics and Photonics. SPIE, 2008, pp. 62 –73. DOI: 10.1117/12.766355. URL: https://doi.org/10.1117/12.766355.

[47] Duc-Tien Dang-Nguyen et al. "Counter-forensics of median filtering". In: *2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE. 2013, pp. 260–265.

[48] Alessia De Rosa et al. "Second-order statistics analysis to cope with contrast enhancement counter-forensics". In: *IEEE Signal Processing Letters* 22.8 (2015), pp. 1132–1136.

[49] Karen O. Egiazarian et al. "A New Full-reference Quality Metrics Based on Hvs". In: 2006.

[50] M. Elad and M. Aharon. "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries". In: *IEEE Transactions on Image Processing* 15.12 (2006), pp. 3736–3745. ISSN: 1941-0042. DOI: 10.1109/TIP.2006.881969.

[51] A Baltimore police officer accidentally recorded himself planting drugs at a crime scene. https://www.vox.com/policy-and-politics/2017/7/19/15999806/baltimore-police-body-camera.

[52] Crowd control: video analytics and alternative surveillance cameras. https://www.asmag.com/showpost/18609.aspx.

[53] Dashcam video captures lucky escape from high speed crash. http://www.mirror.co.uk/news/world-news/-dashcam-captures-drivers-wheely-lucky-6056025.

[54] Detroit OKs 5 million contract for police body and vehicle cameras. http://www.reuters.com/article/us-detroit-police-cameras-idUSKCN0Y82EB.

[55] Drones Used for Crowd Surveillance. https://trackimo.com/drones-used-crowd-surveillance/.

[56] Growing use of police body cameras raises privacy concerns. http://www.latimes.com/nation/la-na-body-cameras-20140927-story.html.

[57] Meteor Smashes YouTube Record. http://mashable.com/2013/02/19/russian-meteor/youtube-record/#9hOcc7EbCPq7.

[58] Paris police are looking for drones to carry out crowd surveillance. https://www.theverge.com/2016/4/27/11516522/paris-police-drone-surveillance-uefa-euro.

[59] Police body cameras welcomed but concerns raised over privacy and misuse. http://www.abc.net.au/news/2017-05-16/tasmanian-cops-to-begin-wearing-cameras-by-end-of-2017/8529238.

[60] Posted dashcam video on Facebook? In Bavaria you face 300,000 euro fine. https://www.rt.com/news/194236-germany-dashcam-video-fine.

[61] The Rapid Rise of Federal Surveillance Drones Over America. http://www.theatlantic.com/politics/archive/2016/03/the-rapid-rise-of-federal/-surveillance-drones-over-america/473136.

[62] Wei Fan et al. "A variational approach to JPEG anti-forensics". In: *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE. 2013, pp. 3058–3062.

[63] Wei Fan et al. "JPEG anti-forensics using non-parametric DCT quantization noise estimation and natural image statistics". In: *Proceedings of the first ACM workshop on Information hiding and multimedia security*. ACM. 2013, pp. 117–122.

[64] Wei Fan et al. "JPEG anti-forensics with improved tradeoff between forensic undetectability and image quality". In: *IEEE Transactions on Information Forensics and Security* 9.8 (2014), pp. 1211–1226.

[65] A. Fathi, A. Farhadi, and J. M. Rehg. "Understanding egocentric activities". In: *Proc. IEEE ICCV*. 2011.

[66] Alireza Fathi, Jessica K Hodgins, and James M Rehg. "Social interactions: A first-person perspective". In: *Proc. IEEE CVPR*. IEEE. 2012, pp. 1226–1233.

[67] Alireza Fathi, Yin Li, and James M. Rehg. "Learning to Recognize Daily Actions Using Gaze". In: *Proc. IEEE ECCV*. 2012.

[68] Rob Fergus et al. "Removing Camera Shake from a Single Photograph". In: *ACM Trans. Graph.* 25.3 (July 2006).

[69] T. Filler, J. Fridrich, and M. Goljan. "Using sensor pattern noise for camera model identification". In: (2008), pp. 1296–1299.

[70] Luciano Floridi. "Artificial intelligence, deepfakes and a future of ectypes". In: *Philosophy & Technology* 31.3 (2018), pp. 317–321.

[71] G. L. Foresti. "Object recognition and tracking for remote video surveillance". In: *IEEE Transactions on Circuits and Systems for Video Technology* 9.7 (1999), pp. 1045–1062. ISSN: 1051-8215. DOI: 10.1109/76.795058.

[72] J. Fridrich and J. Kodovsky. "Rich Models for Steganalysis of Digital Images". In: *IEEE Transactions on Information Forensics and Security* 7.3 (2012), pp. 868–882.

[73] Ohad Fried et al. "Text-based Editing of Talking-head Video". In: *CoRR* abs/1906.01524 (2019). arXiv: 1906.01524. URL: http://arxiv.org/abs/1906.01524.

[74] Nicholas Gardiner. "Facial re-enactment, speech synthesis and the rise of the Deepfake". In: (2019).

[75] Zeno J. Geradts et al. "Methods for identification of images acquired with digital cameras". In: vol. 4232. 2001, pp. 4232–423–8.

[76] Thomas Gloe et al. "Can we trust digital image forensics?" In: *Proceedings of the 15th ACM international conference on Multimedia*. ACM. 2007, pp. 78–86.

[77] Miroslav Goljan, Jessica Fridrich, and Tomas Filler. *Large scale test of sensor fingerprint camera identification*. 2009.

[78] David Güera and Edward J Delp. "Deepfake video detection using recurrent neural networks". In: *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE. 2018, pp. 1–6.

[79] James Hays and Alexei A. Efros. "im2gps: estimating geographic information from a single image". In: *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2008.

[80] Hu He and L. P. Kondi. "Choice of threshold of the Huber-Markov prior in MAP-based video resolution enhancement". In: *Canadian Conference on Electrical and Computer Engineering 2004*. Vol. 2. 2004, 801–804 Vol.2. DOI: 10.1109/CCECE.2004.1345235.

[81] Y. Hel-Or and D. Shaked. "A Discriminative Approach for Wavelet Denoising". In: *IEEE Transactions on Image Processing* 17.4 (2008), pp. 443–457. ISSN: 1941-0042. DOI: 10.1109/TIP.2008.917204.

[82] M Hirsch et al. "Detection of fetal breathing activity in real-time by means of spectral analysis of fetal heart rate fluctuations". In: *Proceedings of the 2nd World Congress of Perinatal Medicine, Rome, Italy*. Vol. 19. 1993, p. 24.

[83] Yedid Hoshen and Shmuel Peleg. "Egocentric Video Biometrics". In: *CoRR* abs/1411.7591 (2014). URL: http://arxiv.org/abs/1411.7591.

[84] Chih-Chung Hsu et al. "Video forgery detection using correlation of noise residue". In: *2008 IEEE 10th Workshop on Multimedia Signal Processing*. 2008, pp. 170–174.

[85] Y. Hu, C. Jian, and C.T. Li. "Using improved imaging sensor pattern noise for source camera identification". In: *Proceedings of IEEE Int. Conf. Multimedia Expo (ICME)* (2010), pp. 1481–1486.

[86] Y. Hu, B. Yu, and C. Jian. "Source Camera Identification Using Large Components of Sensor Pattern Noise". In: *2009 2nd International Conference on Computer Science and its Applications*. 2009, pp. 1–5.

[87] Y. Hu, B. Yu, and C. Jian. "Source camera identification using large components of sensor pattern noise". In: *Proceedings of Int. Conf. Computer Science Applic. (CSA'09)* 23 (Dec. 2009), pp. 1–5.

[88] Z. Hu et al. "Deblurring Low-light Images with Light Streaks". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017), pp. 1–1. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2017.2768365.

[89] Amos Y Johnson, Jie Sun, and Aaron F Bobick. "Using similarity scores from a small gallery to estimate recognition performance for larger galleries". In: *Analysis and Modeling of Faces and Gestures, 2003. AMFG 2003. IEEE International Workshop on*. IEEE. 2003, pp. 100–103.

[90] C Richard Johnson et al. "Image Processing for Artist Identification - Computerized Analysis of Vincent van Gogh's Painting Brushstrokes". In: *IEEE Signal Processing Magazine* (2008).

[91] Kenneth K. Y. Wong Kai San Choi Edmund Y. Lam. *Source camera identification using footprints from lens aberration*. 2006.

[92] X. Kang, Y. Li, and J. Huang. "Enhancing source camera identification performance with a camera reference phase sensor pattern noise". In: *Information Forensics and Security, IEEE Transaction on* (2012), pp. 393–402.

[93] X. Kang et al. "Enhancing Source Camera Identification Performance With a Camera Reference Phase Sensor Pattern Noise". In: *IEEE Transactions on Information Forensics and Security* 7.2 (2012), pp. 393–402.

[94] Xiangui Kang et al. "A context-adaptive SPN predictor for trustworthy source camera identification". In: *EURASIP Journal on Image and Video Processing* 2014.1 (2014), p. 19.

[95] Scott Kangas, Olga Mendoza-Schrock, and Andrew Freeman. "Applying manifold learning to vehicle classification using vibrometry signatures". In: vol. 8751. 2013, 87510G–87510G–9. DOI: 10.1117/12.2018904. URL: http://dx.doi.org/10.1117/12.2018904.

[96] J Karin, M Hirsch, and S Akselrod. "An estimate of fetal autonomic state by spectral analysis of fetal heart rate fluctuations". In: *Pediatric research* 34.2 (1993), p. 134.

[97] Tero Karras, Samuli Laine, and Timo Aila. "A Style-Based Generator Architecture for Generative Adversarial Networks". In: *CoRR* abs/1812.04948 (2018). arXiv: 1812.04948. URL: http://arxiv.org/abs/1812.04948.

[98] Nitin Khanna et al. "A survey of forensic characterization methods for physical devices". In: *digital investigation* 3 (2006), pp. 17–28.

[99] M. Kharrazi, H. T. Sencar, and N. Memon. "Blind source camera identification". In: *Image Processing, 2004. ICIP '04. 2004 International Conference on*. Vol. 1. 2004, 709–712 Vol. 1.

[100] Dong-Hyun Kim and Hae-Yeoun Lee. "Image manipulation detection using convolutional neural network". In: *International Journal of Applied Engineering Research* 12.21 (2017), pp. 11640–11646.

[101] Dongkyu Kim et al. "Median filtered image restoration and anti-forensics using adversarial networks". In: *IEEE Signal Processing Letters* 25.2 (2017), pp. 278–282.

[102] S. J. Kim et al. "A New In-Camera Imaging Model for Color Computer Vision and Its Application". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.12 (2012), pp. 2289–2302.

[103] Jan Kodovsky, Jessica Fridrich, and Vojtěch Holub. "Ensemble classifiers for steganalysis of digital media". In: *IEEE Transactions on Information Forensics and Security* 7.2 (2012), pp. 432–444.

[104] P. Korus and J. Huang. "Multi-Scale Analysis Strategies in PRNU-Based Tampering Localization". In: *IEEE Transactions on Information Forensics and Security* 12.4 (2017), pp. 809–824.

[105] D. Krishnan, T. Tay, and R. Fergus. "Blind deconvolution using a normalized sparsity measure". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. 2011, pp. 233–240. DOI: 10.1109/CVPR.2011.5995521.

[106] Dilip Krishnan and Rob Fergus. "Fast image deconvolution using hyper-Laplacian priors". In: *Advances in neural information processing systems*. 2009, pp. 1033–1041.

[107] K. Kurosawa, K. Kuroki, and N. Saitoh. "CCD fingerprint method-identification of a video camera from videotaped images". In: *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*. Vol. 3. 1999, 537–540 vol.3.

[108] Chun-Wing Kwok, Oscar C Au, and Sung-Him Chui. "Alternative anti-forensics method for contrast enhancement". In: *Proc. of International Workshop on Digital Watermarking*. Springer. 2011, pp. 398–410.

[109] Eric C. Larson and Damon M. Chandler. "Most apparent distortion: full-reference image quality assessment and the role of strategy". In: *J. Electronic Imaging* 19 (2010), p. 011006.

[110] Fabien Lauer and Yann Guermeur. "MSVMpack: a multi-class support vector machine package". In: *Journal of Machine Learning Research* 12.Jul (2011), pp. 2293–2296.

[111] Anat Levin et al. "Image and Depth from a Conventional Camera with a Coded Aperture". In: *ACM Trans. Graph.* 26.3 (July 2007).

[112] C. T. Li. "Unsupervised classification of digital images using enhanced sensor pattern noise". In: *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. 2010, pp. 3429–3432.

[113] C. T. Li and Y. Li. "Color-Decoupled Photo Response Non-Uniformity for Digital Image Forensics". In: *IEEE Transactions on Circuits and Systems for Video Technology* 22.2 (2012), pp. 260–271.

[114] C.T. Li. "Source camera identification using enhanced sensor pattern noise". In: *Image Processing, IEEE Transactions on* 5 (2010), pp. 280–287.

[115] R. Li, C. T. Li, and Y. Guan. "A compact representation of sensor fingerprint for camera identification and fingerprint matching". In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015, pp. 1777–1781.

[116] Stan Z Li. "Markov random field modeling in image analysis". In: (), pp. 43–118.

[117] Yuezun Li and Siwei Lyu. "Exposing deepfake videos by detecting face warping artifacts". In: ().

[118] H. Y. Lin and J. Y. Wei. "A Street Scene Surveillance System for Moving Object Detection, Tracking and Classification". In: *2007 IEEE Intelligent Vehicles Symposium*. 2007, pp. 1077–1082. DOI: 10.1109/IVS.2007.4290260.

[119] X. Lin and C. T. Li. "Preprocessing Reference Sensor Pattern Noise via Spectrum Equalization". In: *IEEE Transactions on Information Forensics and Security* 11.1 (2016), pp. 126–140.

[120] Xufeng Lin, Chang-Tsun Li, and Yongjian Hu. "Exposing image forgery through the detection of contrast enhancement". In: *Proc. of IEEE International Conference on Image Processing*. 2013, pp. 4467–4471.

[121] B. b. Liu et al. "On classification of source cameras: A graph based approach". In: *2010 IEEE International Workshop on Information Forensics and Security*. 2010, pp. 1–5.

[122] Bo Liu and Chi-Man Pun. "Deep fusion network for splicing forgery localization". In: *The European Conference on Computer Vision (ECCV) Workshops*. 2018.

[123] David G Lowe. "Distinctive image features from scale-invariant keypoints". In: *International journal of computer vision* 60.2 (2004), pp. 91–110.

[124] Wenmiao Lu and Yap-Peng Tan. "Color filter array demosaicking: new method and performance measures". In: *IEEE transactions on image processing* 12.10 (2003), pp. 1194–1210.

[125] Jan Lukas, Jessica Fridrich, and Miroslav Goljan. "Digital camera identification from sensor pattern noise". In: *IEEE Transactions on Information Forensics and Security* 1.2 (2006), pp. 205–214.

[126] J. Lukáš, J. Fridrich, and M. Goljan. "Digital camera identification from sensor pattern noise". In: *Information Forensics and Security, IEEE Transactions on* 1 (June 2006), pp. 205–214.

[127] Bingpeng Ma, Yu Su, and Frédéric Jurie. "Local descriptors encoded by fisher vectors for person re-identification". In: *Proc. Springer ECCV*. 2012.

[128] Laurens van der Maaten and Geoffrey Hinton. "Visualizing data using t-SNE". In: *Journal of Machine Learning Research* 9.Nov (2008), pp. 2579–2605.

[129] Babak Mahdian and Stanislav Saic. "A bibliography on blind methods for identifying image forgery". In: *Signal Processing: Image Communication* 25.6 (2010), pp. 389 –399. ISSN: 0923-5965. DOI: https://doi.org/10.1016/j.image.2010.05.003.

[130] J. Mairal et al. "Non-local sparse models for image restoration". In: *2009 IEEE 12th International Conference on Computer Vision*. 2009, pp. 2272–2279. DOI: 10.1109/ICCV.2009.5459452.

[131] J. Mantyjarvi et al. "Identifying users of portable devices from gait pattern with accelerometers". In: *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.* Vol. 2. 2005, ii/973–ii/976 Vol. 2.

[132] Marie-Helen Maras and Alex Alexandrou. "Determining authenticity of video evidence in the age of artificial intelligence and in the wake of Deepfake videos". In: *The International Journal of Evidence & Proof* (2018), p. 1365712718807226.

[133] Sébastien Marcel. "BEAT–biometrics evaluation and testing". In: *Biometric technology today* 2013.1 (2013), pp. 5–7.

[134] Francesco Marra et al. "Blind PRNU-based image clustering for source identification". In: *IEEE Transactions on Information Forensics and Security* 12.9 (2017), pp. 2197–2211.

[135] Stacy MC, Augustine KE, and Robb RA. "Image BOSS: An image database system designed for research." In: *Journal of Digital Imaging* (1997), 10(Suppl 1):56–59. DOI: 10.1007/BF03168658.

[136] Joel McCall et al. "A collaborative approach for human-centered driver assistance systems". In: *Proc. IEEE ITS.* 2004.

[137] A. Mehrish, A. V. Subramanyam, and M. Kankanhalli. "Multimedia signatures for vehicle forensics". In: *Proc. IEEE ICME*. 2017.

[138] Hiroshi Murase and Rie Sakai. "Moving object recognition in eigenspace representation: gait analysis and lip reading". In: *Pattern Recognition Letters* 17.2 (1996), pp. 155 –162.

[139] Thomas P O'Rourke and Robert L Stevenson. "Improved image decompression for reduced transform coding artifacts". In: *IEEE Transactions on Circuits and Systems for Video Technology* 5.6 (1995), pp. 490–499.

[140] Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton van den Hengel. "Learning to rank in person re-identification with metric ensembles". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 1846–1855.

[141] Sangkeun Park et al. "Motives and Concerns of Dashcam Video Sharing". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: ACM, 2016, pp. 4758–4769. ISBN: 978-1-4503-3362-7. DOI: 10.1145/2858036.2858581. URL: http://doi.acm.org/10.1145/2858036.2858581.

[142] F. Pascal et al. "Parameter Estimation For Multivariate Generalized Gaussian Distributions". In: *Image Processing, IEEE Transactions on.* 61 (Nov. 2013), pp. 5960–5971.

[143] T. Pevny, P. Bas, and J. Fridrich. "Steganalysis by Subtractive Pixel Adjacency Matrix". In: *IEEE Transactions on Information Forensics and Security* 5.2 (2010), pp. 215–224. ISSN: 1556-6013.

[144] Tomas Pevny and Jessica Fridrich. "Merging Markov and DCT features for multi-class JPEG steganalysis". In: *Proc. of Security, Steganography, and Watermarking of Multimedia Contents IX*. Vol. 6505. International Society for Optics and Photonics. 2007, p. 650503.

[145] N. Ponomarenko et al. "Modified image visual quality metrics for contrast change and mean shift accounting". In: *2011 11th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. 2011, pp. 305–311.

[146] A. C. Popescu and H. Farid. "Exposing digital forgeries by detecting traces of resampling". In: *IEEE Transactions on Signal Processing* 53.2 (2005), pp. 758–767. ISSN: 1053-587X. DOI: 10.1109/TSP.2004.839932.

[147] J. Portilla et al. "Image denoising using scale mixtures of Gaussians in the wavelet domain". In: *IEEE Transactions on Image Processing* 12.11 (2003), pp. 1338–1351. ISSN: 1941-0042. DOI: 10.1109/TIP.2003.818640.

[148] Yanjun Qi, Judith Klein-Seetharaman, and Ziv Bar-Joseph. "Random forest similarity for protein-protein interaction prediction from multiple sources". In: *Biocomputing 2005*. World Scientific, 2005, pp. 531–542.

[149] T. Qiao et al. "Source camera identification based on raw images". In: *Proceedings of ICIP' 15* (2015), pp. 3812–3816.

[150] Xiaoqing Qiu et al. "A Universal Image Forensic Strategy Based on Steganalytic Model". In: *Proc. of the 2nd ACM Workshop on Information Hiding and Multimedia Security*. 2014.

[151] H. Ravi, A. V. Subramanyam, and S. Emmanuel. "ACE-An Effective Anti-forensic Contrast Enhancement Technique". In: *IEEE Signal Processing Letters* 23.2 (2016), pp. 212–216. ISSN: 1070-9908. DOI: 10.1109/LSP.2015.2509477.

[152] Hareesh Ravi, A. V. Subramanyam, and Sabu Emmanuel. "Forensic Analysis of Linear and Nonlinear Image Filtering Using Quantization Noise". In: *ACM Trans. Multimedia Comput. Commun. Appl.* 12.3 (Mar. 2016), 39:1–39:23. ISSN: 1551-6857.

[153] Hareesh Ravi, A. V. Subramanyam, and Sabu Emmanuel. "Spatial domain quantization noise based image filtering detection". In: *Proc. of IEEE International Conference on Image Processing*. 2015, pp. 1180–1184.

[154] M. A. Robertson and R. L. Stevenson. "DCT quantization noise in compressed images". In: *IEEE Transactions on Circuits and Systems for Video Technology* 15 (2005), pp. 27–38.

[155] S. Roth and M. J. Black. "Fields of Experts: a framework for learning image priors". In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*. Vol. 2. 2005, pp. 860–867.

[156] Stefan Roth and Michael J. Black. "Fields of Experts". In: *International Journal of Computer Vision* 82.2 (2009), p. 205. ISSN: 1573-1405. DOI: 10.1007/s11263-008-0197-6. URL: https://doi.org/10.1007/s11263-008-0197-6.

[157] Ronald Salloum, Yuzhuo Ren, and C-C Jay Kuo. "Image splicing localization using a multi-task fully convolutional network (MFCN)". In: *Journal of Visual Communication and Image Representation* 51 (2018), pp. 201–209.

[158] R. R. Schultz and R. L. Stevenson. "A Bayesian approach to image expansion for improved definition". In: *IEEE Transactions on Image Processing* 3.3 (1994), pp. 233–242. ISSN: 1057-7149. DOI: 10.1109/83.287017.

[159] Richard R Schultz and Robert L Stevenson. "Extraction of high-resolution frames from video sequences". In: *IEEE transactions on image processing* 5.6 (1996), pp. 996–1011.

[160] Wuyang Shan et al. "Robust contrast enhancement forensics based on convolutional neural networks". In: *Signal Processing: Image Communication* 71 (2019), pp. 138–146.

[161] H. R. Sheikh and A. C. Bovik. "Image information and visual quality". In: *IEEE Transactions on Image Processing* 15.2 (2006), pp. 430–444. ISSN: 1057-7149. DOI: 10.1109/TIP.2005.859378.

[162] Ashley Smith et al. "An end-to-end vechicle classification pipeline using vibrometry data". In: *SPIE Defense+ Security*. International Society for Optics and Photonics. 2014, 90790O–90790O.

[163] Ashley Smith et al. "Remote vibrometry vehicle classification". In: *Proc. SPIE* 9464 (2015), 94640S–94640S–13.

[164] L. Sommer, T. Schuchert, and J. Beyerer. "Comprehensive Analysis of Deep Learning based Vehicle Detection in Aerial Images". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2018), pp. 1–1. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2018.2874396.

[165] M. J. Sorrell. "Digital camera source identification through JPEG quantisation". In: *Multimedia Forensics and Security Hershey NY USA: Information Science Reference* (2008), pp. 291–313.

[166] Matthew Stamm and KJ Ray Liu. "Blind forensics of contrast enhancement in digital images". In: *Proc. of IEEE International Conference on Image Processing*. 2008, pp. 3112–3115.

[167] Matthew C Stamm and KJ Ray Liu. "Anti-forensics of digital image compression". In: *IEEE Transactions on Information Forensics and Security* 6.3 (2011), pp. 1050–1065.

[168] Matthew C Stamm and KJ Ray Liu. "Forensic detection of image manipulation using statistical intrinsic fingerprints". In: *IEEE Transactions on Information Forensics and Security* 5.3 (2010), pp. 492–506.

[169] Matthew C Stamm and KJ Ray Liu. "Forensic estimation and reconstruction of a contrast enhancement mapping." In: *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*. 2010, pp. 1698–1701.

[170] Rob Sumner. "Processing raw images in matlab". In: (2014). URL: https://rcsumner.net/raw_guide/RAWguide.pdf.

[171] Jee-Young Sun et al. "A novel contrast enhancement forensics based on convolutional neural networks". In: *Signal Processing: Image Communication* 63 (2018), pp. 149–160.

[172] Jian Sun et al. "Learning a convolutional neural network for non-uniform motion blur removal". In: *Proc. IEEE CVPR*. 2015.

[173] Y. Sutcu et al. "Improvements on sensor noise based source camera identification". In: *Proceedings of IEEE Int.Conf. Multimedia Expo* (2007), pp. 24–27.

[174] Charles Sutton, Andrew McCallum, et al. "An introduction to conditional random fields". In: *Foundations and Trends® in Machine Learning* 4.4 (2012), pp. 267–373.

[175] A. Swaminathan, M. Wu, and K. J. R. Liu. "Nonintrusive component forensics of visual sensors using output images". In: *IEEE Transactions on Information Forensics and Security* 2.1 (2007), pp. 91–106.

[176] Thanh Hai Thai, Florent Retraint, and RÃl'mi Cogranne. "Camera model identification based on DCT coefficient statistics". In: *Digital Signal Processing* 40.Supplement C (2015), pp. 88 –100. ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2015.01.002.

[177] Thanh Hai Thai, Florent Retraint, and RÃl'mi Cogranne. "Camera model identification based on the generalized noise model in natural images". In: *Digital Signal Processing* 48.Supplement C (2016), pp. 285 –297. ISSN: 1051-2004. DOI: https://doi.org/10.1016/j.dsp.2015.10.002.

[178] Christopher Thomas and Adriana Kovashka. "Who's Behind the Camera? Identifying the Authorship of a Photograph". In: *CoRR* (2015). URL: http://arxiv.org/abs/1508.05038.

[179] Amel Tuama, Frédéric Comby, and Marc Chaumont. "Camera model identification with the use of deep convolutional neural networks". In: *2016 IEEE International workshop on information forensics and security (WIFS)*. IEEE. 2016, pp. 1–6.

[180] L. T. Van, S. Emmanuel, and M. S. Kankanhalli. "Identifying Source Cell Phone using Chromatic Aberration". In: *2007 IEEE International Conference on Multimedia and Expo*. 2007, pp. 883–886.

[181] Q. Wang, J. Wan, and X. Li. "Robust Hierarchical Deep Learning for Vehicular Management". In: *IEEE Transactions on Vehicular Technology* (2018), pp. 1–1. ISSN: 0018-9545. DOI: 10.1109/TVT.2018.2883046.

[182] Xinyi Wang et al. "Detection and localization of image forgeries using improved mask regional convolutional neural network". In: ().

[183] Z. Wang, E. P. Simoncelli, and A. C. Bovik. "Multiscale structural similarity for image quality assessment". In: *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*. Vol. 2. 2003, 1398–1402 Vol.2. DOI: 10.1109/ACSSC.2003.1292216.

[184] Zhou Wang et al. "Image quality assessment: from error visibility to structural similarity". In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612. ISSN: 1057-7149. DOI: 10.1109/TIP.2003.819861.

[185] O. Whyte et al. "Non-uniform Deblurring for Shaken Images". In: *International Journal of Computer Vision* 98.2 (2012), pp. 168–186.

[186] G. Wu, X. Kang, and K. J. R. Liu. "A context adaptive predictor of sensor pattern noise for camera source identification". In: *2012 19th IEEE International Conference on Image Processing*. 2012, pp. 237–240.

[187] Bingchao Xu et al. "Source camera identification from image texture features". In: *Neurocomputing* 207.Supplement C (2016), pp. 131 –140.

ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2016.05.012.

[188] Robin Elizabeth Yancey. "Bilinear Faster RCNN with ELA for Image Tampering Detection". In: *CoRR* abs/1904.08484 (2019). arXiv: 1904.08484. URL: http://arxiv.org/abs/1904.08484.

[189] Pengpeng Yang et al. "Robust Contrast Enhancement Forensics Using Convolutional Neural Networks". In: *arXiv preprint arXiv:1803.04749* (2018).

[190] Wotao Yin et al. "Bregman iterative algorithms for L1-minimization with applications to compressed sensing". In: *SIAM Journal on Imaging Sciences* 1.1 (2008), pp. 143–168.

[191] Yuan Yuan, Zhiyu Jiang, and Qi Wang. "Video-based road detection via online structural learning". In: *Neurocomputing* 168 (2015), pp. 336–347. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.05.092. URL: http://www.sciencedirect.com/science/article/pii/S0925231215007900.

[192] Egor Zakharov et al. "Few-Shot Adversarial Learning of Realistic Neural Talking Head Models". In: *CoRR* abs/1905.08233 (2019). arXiv: 1905.08233. URL: http://arxiv.org/abs/1905.08233.

[193] Hui Zeng et al. "Countering anti-forensics of median filtering". In: *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*. 2014, pp. 2704–2708.

[194] L. Zhang et al. "FSIM: A Feature Similarity Index for Image Quality Assessment". In: *IEEE Transactions on Image Processing* 20.8 (2011), pp. 2378–2386. ISSN: 1057-7149. DOI: 10.1109/TIP.2011.2109730.

[195] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. "Unsupervised salience learning for person re-identification". In: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE. 2013, pp. 3586–3593.

[196] J. Zhu, S. Liao, and S. Z. Li. "Multicamera Joint Video Synopsis". In: *IEEE Transactions on Circuits and Systems for Video Technology* 26.6 (2016), pp. 1058–1069. ISSN: 1051-8215. DOI: 10.1109/TCSVT.2015.2430692.

[197] Karel Zuiderveld. "Graphics Gems IV". In: ed. by Paul S. Heckbert. San Diego, CA, USA: Academic Press Professional, Inc., 1994. Chap. Contrast Limited Adaptive Histogram Equalization, pp. 474–485. ISBN: 0-12-336155-9. URL: http://dl.acm.org/citation.cfm?id=180895.180940.