



MANIPULATION DETECTION IN UNCOMPRESSED VIDEO

NIDHAL MUHSIN HAZZAA

FEBRUARY 2016

MANIPULATION DETECTION IN UNCOMPRESSED VIDEO

**A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES OF
ÇANKAYA UNIVERSITY**

**BY
NIDHAL MUHSIN HAZZAA**

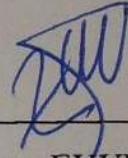
**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF
MATHEMATICS AND COMPUTER SCIENCE
INFORMATION TECHNOLOGY PROGRAM**

FEBRUARY 2016

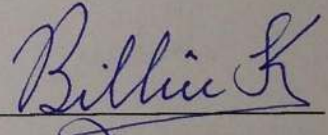
Title of the Thesis : **MANIPULATION DETECTION IN UNCOMPRESSED VIDEO**

Submitted by **Nidhal Muhsin Hazzaa**

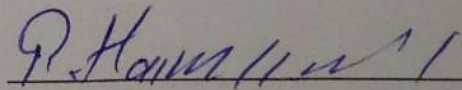
Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.


Prof. Dr. Halil Tanyer EYYUBOĞLU
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.


Prof. Dr. Billur KAYMAKÇALAN
Head of Department

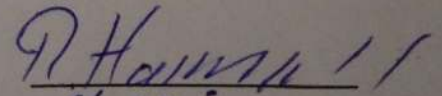
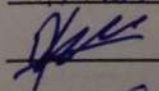
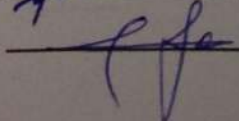
This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.


Assist. Prof. Dr. REZA HASSANPOUR
Supervisor

Examination Date: 04.02.2016

Examining Committee Members

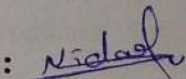
Assist. Prof. Dr. REZA HASSANPOUR (Çankaya Univ.)
Assist. Prof. Dr. Abdül Kadir GÖRÜR (Çankaya Univ.)
Assoc. Prof. Dr. Fahad JARAD (UTAA)

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Nidhal HAZZAA

Signature : 

Date : 04.02.2016

ABSTRACT

MANIPULATION DETECTION IN UNCOMPRESSED VIDEO

HAZZAA, Nidhal

M.Sc., Department of Mathematics and Computer Science

Supervisor: Assist. Prof. Dr. REZA HASSANPOUR

FEBRUARY 2016, 56 pages

In recent years due to advancement in video and image editing tools it has become increasingly easy to modify content of multimedia. Doctored videos are very difficult to identify through visual examination as artifacts left behind by processing steps are subtle and cannot be easily captured visually. Therefore, the integrity of digital videos cannot be taken for granted and these are not readily acceptable as a proof-of-evidence in a court-of-law. Hence, identifying the authenticity of videos has become an important field of information security. This thesis presents an approach to detect and temporally localize video forgery, based on the correlation of noise residual using the Discrete Wavelet Transformation (DWT) algorithm for de-noising. The proposed algorithm is tested on public datasets such as SULFA are used for performance evaluation. The results show that the approach is effective against manipulation techniques. In addition, it detects and localizes tampered frames in a video with high accuracy.

Keywords: Discrete Wavelet Transformation, Video Forensic, Video Forgery, correlation of noise residue.

ÖZ

SIKIŞTIRIMAMIŞ VIDEONIN MANİPÜLASYON TESPİT

HAZZAA, Nidhal

Yüksek Lisans, Matematik Bilgisayar Anabilim Dalı

Tez Yöneticisi: Yrd. Doç. Dr. REZA HASSANPOUR

Şubat 2016, 56 sayfa

Son yıllarda video ve resim düzenleme araçlarının ilerlemesi nedeniyle çoklu ortam içeriğini değiştirmek giderek daha kolay hale gelmiştir. İşleme adımlarından kalan eserler gizli ve görsel olarak kolayca fark edilemediğinden, işlenmiş videoların görsel muayene ile tespit edilmesi çok zordur. Bu yüzden, dijital video bütünlüğü hafife alınabilir ve bunlar mahkemede kanıt olarak kolayca kabul edilemez. Dolayısıyla, video orijinalliğini tanımlamak, bilgi güvenliğinin önemli bir alanı haline gelmiştir. Bu tez gürültü kaldırmak için yapılmış Ayrık Dalgacık Dönüşümü (ADD) algoritmasını kullanarak gürültü kalıntısının korelasyonuna dayalı video sahtekârlığını tespit edici ve geçici olarak lokalize edici bir çalışma sunar. Sunulan algoritmanın performans değerlendirilmesi bir kamu veri tabanı olan SULFA'da test edilmiştir. Sonuçlar bu çalışmanın manipülasyon tekniklerine karşı etkili olduğunu göstermektedir. Buna ek olarak, yüksek kaliteli bir videonun işlenmiş karelerini tespit ve lokalize eder.

Anahtar Kelimeler: Ayrık Dalgacık Dönüşümü, Adli Video, Video Sahtekârlığı, Gürültü Kalıntısının Korelasyonu.

ACKNOWLEDGEMENTS

I would like to express my appreciation to my supervisor, Assist.Prof. Dr. REZA HASSANPOUR who gave me unlimited support and valuable guidance. Without him this thesis would not have been completed or written. There are not enough words to express my thanks to you.

Especially I would like to offer my thesis as a gift to the soul of my parents ...

Finally, I would like to extend special gratitude to my family and friends for their endless and continuous encouragement and support throughout the years, which has enabled me to complete this work.

TABLE OF CONTENTS

STATEMENT OF NON PLAGIARISM.....	iii
ABSTRACT.....	iv
ÖZ.....	v
ACKNOWLEDGEMENTS.....	vi
TABLE OF CONTENTS.....	vii
LIST OF FIGURES.....	ix
LIST OF TABLES.....	xii
LIST OF ABBREVIATIONS.....	xiii

CHAPTERS

1. INTRODUCTION.....	1
1.1. Background.....	1
1.2. Video Forensic.....	3
1.3. Motivation	4
1.4 Thesis Objective.....	5
1.5 Thesis organization.....	5
2. LITERATURE REVIEW	6
2.1. Forgery.....	6
2.1.1. Image Forgery.....	7
2.1.2. Image Forgery Detection Techniques.....	8
2.1.2.1. DCT-based algorithms.....	10
2.1.2.2. Algorithms based on PCA.....	11
2.1.2.3. Algorithms based on Log-polar transform.....	11
2.1.2.4. Invariant key-points based algorithms.....	12
2.1.2.5. SVD-based algorithms.....	13
2.1.2.6. Algorithms based on Texture and intensity.....	14
2.1.2.7. Other algorithms.....	15

LIST OF FIGURES

FIGURES

Figure 1	Example from one shot of Manipulated Video.....	1
Figure 2	Classifications Of Forgery Types.....	6
Figure 3	Sample of image forgery showing the original image, the tampered image and the copied regions detection.....	8
Figure 4	Classifications of Image Forgery Detection Methods.....	9
Figure 5	Examples of (A) Original Video Sequences (B) Spatially Tampered Video Sequences (C) Temporally Tampered Video Sequences and (D) Spatio-Temporal Tampered Video Sequences.....	17
Figure 6	Video Forgery Detection Techniques.....	19
Figure 7	Mallat's algorithm display - 3-level decomposition of a signal...	26
Figure 8	Example Mallat's algorithm.....	26
Figure 9	Signal f_1 and its noisy version y	27
Figure 10	Flowchart of the proposed algorithm.....	30
Figure 11	Samples for noisy wavelet coefficients.....	32
Figure 12	First frame after implemented de-noising algorithm (a) original forged image, (b) noise-free image.....	34
Figure 13	Result of noise residual extraction.....	35
Figure 14	Pixel intensity for first block.....	36
Figure 15	Representation of calculating the correlation of the noise residual among the current frame blocks and previous frame blocks.....	37
Figure 16	Comparison of noise correlation values between two neighboring video blocks in forged and non-forged regions.....	39

Figure 17	First forgery block.....	40
Figure 18	Simulation result for video 1, frame 29, 94, (a) Source Video frames (b) Forged video frames (c) Detection and Localization video frames.....	45
Figure 19	Simulation result for video 1, frame 111,153, (a) Source Video frames (b) Forged video frames (c) Detection and Localization video frames.....	46
Figure 20	Simulation result for video 2, frame 4, 9 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	47
Figure 21	Simulation result for video 3, frame 133,153 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frame.....	48
Figure 22	Simulation result for video 3, frame 182,315 (a)Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	49
Figure 23	Simulation result for video 4, frame 8, 59 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	50
Figure 24	Simulation result for video 4, frame 116,136 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	51
Figure 25	Simulation result for video 5, frame 8, 53 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	52
Figure 26	Simulation result for video 5, frame 118,168 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	53

Figure 27	Simulation result for video 6, frame 2, 62 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	54
Figure 28	Simulation result for video 6, frame 132,198 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.....	55

LIST OF TABLES

TABLES

Table 1	Dataset Details.....	42
Table 2	Correlation values for video 1, blocks from frame (5, 10, 15, 20, 25).....	43
Table 3	Correlation values for video 1, blocks from frame (50, 55, 60, 65, 70).....	43
Table 4	Correlation values for video 1, blocks from frame (110, 120, 130, 140,150).....	44
Table 5	Correlation values for video 1, blocks from frame (160, 170, 180, 190, 200).....	44

LIST OF ABBREVIATIONS

JPEG	Joint Photographic Experts Group
BMP	BitMaP
GOP	Group Of Picture
MPEG	Moving Picture Experts Group
DWT	Discrete Wavelet Transformation
GMM	Gaussian Mixture Model
DCT	Discrete Cosine Transform
PCA	Principal Component Analysis
SIFT	ScaleInvariant Feature Transform
SURF	Speeded Up Robust Features
RANSAC	RANdom SAmple Consensus
SVD	Singular Value Decomposition
LLE	Locally Linear Embedding
HOG	Histogram of Oriented Gradients
VPF	Variation Prediction Footprint
MCEA	Motion-Compensated Edge Artifact
ML	Maximum Likelihood
MMSE	Minimum Mean Squared Error
AWGN	Additive White Gaussian Noise
MSE	Mean-Squared Error

CHAPTER 1

INTRODUCTION

1. 1 Background

Images and videos are significant instruments of communication and expression in today's world. Videos especially are widely used in a number of areas such as surveillance, medical imaging, the movie industry, journalism, home videos and so on. Although tampering with video is relatively difficult, but in recent times, in consequence of progression in network technologies, advanced image and video editing software rapid growth of digital devices with low cost multimedia devices, it has become very easy to produce and modify digital videos with increasing sophistication. Digital multimedia integrity can now be simply tampered, synthesized and manipulated in various ways without leaving any visible clues [1]. For example, Figure 1 displays a frame from a Russian TVs show in 2007. In that program, a famous political analyst named Mikhail G. Delyagin made some tart remarks about Vladimir V. Putin. Later, when the program was broadcast, not only his remarks cut, he was also erased from the show, although of the fact that the technicians ignored to remove his legs from one shot.



Figure 1 Example from one shot of manipulated video.

1.2 Video Forensics

The content of digital videos can no longer be taken for granted. It has become difficult to recognize the difference between an original video and a doctored one. This phenomenon leads to serious consequences, loss of confidence and the creation of false beliefs in many applications in the real world; therefore, there is an increasing displeasure and mistrust about the authenticity of these videos and such videos can no longer be presented as proof-of-evidence in a court-of-law [1]. This has made video forensics which evaluates the authenticity of a video and proves its trustworthiness become an influential and interesting field. Video forensics can be fall into two categories:

a) Active forensics primarily concentrates on data concealment and hashing techniques such as watermarking [2] and hashing [3]. According to watermarking techniques multimedia data is a communication channel. An embedded watermark, perhaps imperceptible too, contains either a certain producer ID or some content-related codes that are used for authentication [4].

In hashing, by using appropriate algorithm image identifiers or image hash functions are produced, and used for image authentication. However, these approaches need a source camera or video at the time of generation, which makes these approaches limited in terms of practicality. Most video capturing devices do not have a built-in watermarking and signature accounting embedded module, which may also have an opposite effect on video quality, hence these may not be preferred.

b) Passive forensics concentrates on the essential characteristics of digital media or an acquisition device. Passive video forgery detection approaches are used widely in multimedia security, pattern recognition and information security. They are used to detect and localize tampering without depending on any prior information. A passive approach is a better choice to detect forgery when there is no prior information of the source media and acquisition device [5]. Major of the passive approach algorithms detect forgery with localizing the tampered regions.

Localization of suspected areas is a major issue in multimedia forensics. The ability to determine the area of suspicion of an image/video allows one to provide a convincing explanation about any suspected tampering. For example, when a person in an image/video has been successfully connected, it serves as the basis of information for experts to extract more information from the image and conduct an in-depth examination. An expert of local images/videos can compare and find the photo source in a database. If it is successfully found, it not only strengthens the manipulation (because a video is obviously a source of strong evidence), it also leads to further study. For example, time and location of the video source and video manipulation can be used to determine the state of any manipulation. If there are two videos at the same location (for example, a tourist spot) at about the same time, they may be manipulated for aesthetic purposes with the aim to create a better video. In this case, it means the video does not change. Such negotiations can often be categorized as harmless. However, if time and space are clearly distinct, it is more likely that the video has been manipulated in order to create the false belief of production. Localization of modification areas may also be used to link local content in video manipulation resources [11]. Localization of suspected areas significantly relies on imposed place restrictions. The inherent assumption is that the content is connected to interconnect often rather sparse. Connected pixels or blocks in the foreground region must share the same symptoms (e.g., light, device features, or convert index distribution).

Detecting copy-move in an image necessitates extensive searches of local pattern or region matches. In this thesis, we focus on passive forensics. We study the problem of detection and temporal localization of forged regions which are produced from removing objects from videos, and present an effective and robust approach based on a correlation of noise residue. Our approach targeted on video sequences tampered by copy-move techniques which are based on copy a part from frame and paste it into another frame with maintaining on temporal coherence between successive frames to create a plausible tampered video. It takes temporal domain into account

then tampered (add or hide) regions by replacing it with the similar areas from the neighbor frame [6].

One preliminary idea that one gets to detect copy-paste forgery is partitioned video into frames then breaking each frame spatially into blocks of size $n \times n$ by analyzing the temporal correlation of block-level noise residual can locate the tampered areas of a video. Our approach does not need to precollect and pretrain. The noise residue information can be simply extracted from a specific video and be legalized depending on the statistics of the noise residue for the given video cameras. In this work, we have targeted the popular and efficient detection techniques proposed by Hsu, et al [60]. Preprocessing is done by extracting correlation values of noise residuals as a feature for classification of the block-level. A Gaussian mixture model (GMM) is modeled by the distribution of the correlation of temporal noise residual in a tampered video. The bottom-up algorithm is applied for locating the tampered regions of a video built on a block-level temporal noise correlation.

1.3 Motivation

Video forgery is the process of editing, rearranging, adding objects or removing unwanted objects from a video (inpainting) or filling missing or damaged parts of a video sequence with visually plausible information [7]. It has attracted a great deal of attention in recent years because a video is obviously a source of strong evidence and the powerful ability of these techniques to restore damaged videos. However, this video retrieval completion technique can also be used as a major forgery tool to implement malicious changes in videos, such as object-removal and photo-montages [10] as illustrated in Figure 1, showing multimedia content plausibly forged. The existing copy-move algorithms may not be easily extended as the performance of an algorithm depends on the size of any forged patches. Furthermore, sources of copied information may lie in different frames of the video and they may be non-continuous. This means that an object can be filled by a set of multiple small parts located in different places in different frames of the same video. Hence, inpainting forgery is

more difficult to detect than other forgery types and therefore poses a challenging research problem. The existing techniques can detect some of the popular forgery approaches [6] [9]. These include inpainting methods to fill in missing background and moving foreground of real-time videos captured by moving or still camera while simultaneously maintaining spatial and temporal coherence. However, these detection techniques do not perform well with other state-of-the-art inpainting techniques [8] which rely on the optimization of a global, patch-based function. This in turn has necessitated the need for research in the field of video inpainting detection and localization.

1.4 Thesis Objective

This work targets two popularly used correlation techniques for forgery detection.

Thesis objectives are:

1. Given an input video, determine whether it is an authentic video or a manipulated video.
2. For a tampered video, detect and localize any forged regions.

1.5 Thesis organization

Chapter 2 discusses Related Work and presents a literature review.

Chapter 3 proposes an algorithm to detect tampering in videos also in addition to temporally localizing forged frames in a video.

Chapter 4 presents experimental details of the performed experiments and covers the results of the proposed algorithm.

Chapter 5 offer discussions, limitations and future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Forgery

Due to the popularity of low-cost and high-resolution digital cameras, digital media plays an increasingly important role in our daily life. However, according to the professional digital image and video editing software can easily manipulate digital media without leaving clues of visible change, it has become a serious social problem as to how much of their content can be trusted, especially regarding whether it is valid or can be used as evidence in court, for insurance claims and for scientific fraud manipulation. According to a number of statistics [30], in a magazine, as many as 20 percent of accepted manuscripts contain inappropriate data manipulation and 1% of them contain fraudulent manipulation. As a result, when the images or videos are tampered for the purpose of falsehood is false, may result in loss limitless. To combat this problem, digital image and video forensics has grown as a new study field showing how digital manipulations of images are filed. Digital media forgery can be classified into two categories, namely digital image forgery and digital video forgery, Figure 2 display forgery type classification.

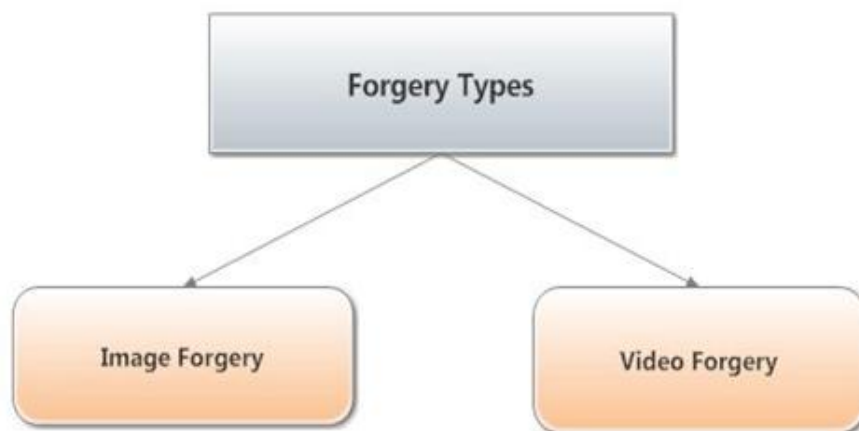


Figure 2 Classifications of Forgery Types

2.1.1 Image Forgery

Manipulation of images due to easy access to advanced powerful image editing software tools and hardware computing has become very easy. Several types of counterfeits can be created and in modern years, counterfeit detection techniques that use passive images have become a contentious field of research [15], [16]. One of the most popular types of counterfeit copy-paste images (or copy move or simulation) is spoofing, where a section of an image is copied and moved onto another part, thus hiding the content of the image in the second area. It can be used to hide an object hidden that is unwanted or it can increment the number of regions in the image appearing there. In spite of the simple translation may be enough in numerous issues, extra treatment and manipulation often happen for better concealment. This may be carried out by rotation, noise addition, scaling, lossy compression and opaque, among other techniques. Consequently, in order to be able to discover such fraudulent manipulations reliably, several techniques have been recently suggested that stronger efforts to some of these developments. As fake copy-pasting is more compelling, devising techniques that can still be manipulated into areas such as identification and disclosure. The fake copy of hidden in an image area covering a different area of the image. The most obvious way to identify areas in the same image will be copied and pasted to investigate tiny group or blocks of pixels to play over the image. However, there are two main issues with this approach. In the first stage, it would be a computationally intensive approach, as it is impossible to increase the size of the implemented blocks (or other shapes) of the Pixel Image Converter. Second approach is used in the case of minor changes such as the image compression few failures or adding noise. An example is shown in Figure 3 of the counterfeit copy.



Figure 3 Sample of tampered image with the detection of manipulated regions.

2.1.2 Image forgery detection techniques

Commonly, two techniques are used for image manipulation detection: active techniques and passive techniques. The active techniques are categorized into two classes; the first one is based on digital watermarking technique and the second being based on digital signatures. The passive detection technique, also referred to as digital image forensics, detects the forged images without employing any information from the source images. The passive methods are classified into two classes; namely dependent and independent forgery types. The independent techniques are useful for detecting three various types of tampering: explored as compression, re-sampling and inconsistencies. The dependent techniques are purposed for distinctive kinds of tampering, for example copy-paste which involve forgery of single image) and (forgeries of multiple image) named image splicing [38], as shown in Figure 4 The smooth and activity of copy-paste tampering makes it

the popular type of manipulation which is used to tamper the objects of an image [14]. Copy-move forgery is the process of copying an area from a picture and pasting it into another place on the same picture. The goal is to conceal unwanted objects or replace them. Because the copied area comes from the same image, which has the same significant properties which are compatible with the rest of the image, such as dynamic range and the color palette. Copy-paste tampering defines a correlation among the original image and the tampered image. This correlation can be used as a fundamental for many detection algorithms of this type of forgery [19]. There are many methods to detect a passive copy-paste forgery in a digital image. The main variance between the existing techniques is the size and type of the features which are used for matching the image blocks. The existing methods are classified depending on the extracted features the similarities of the blocks are testing. In the next sections, image forgery detection algorithms are presented.

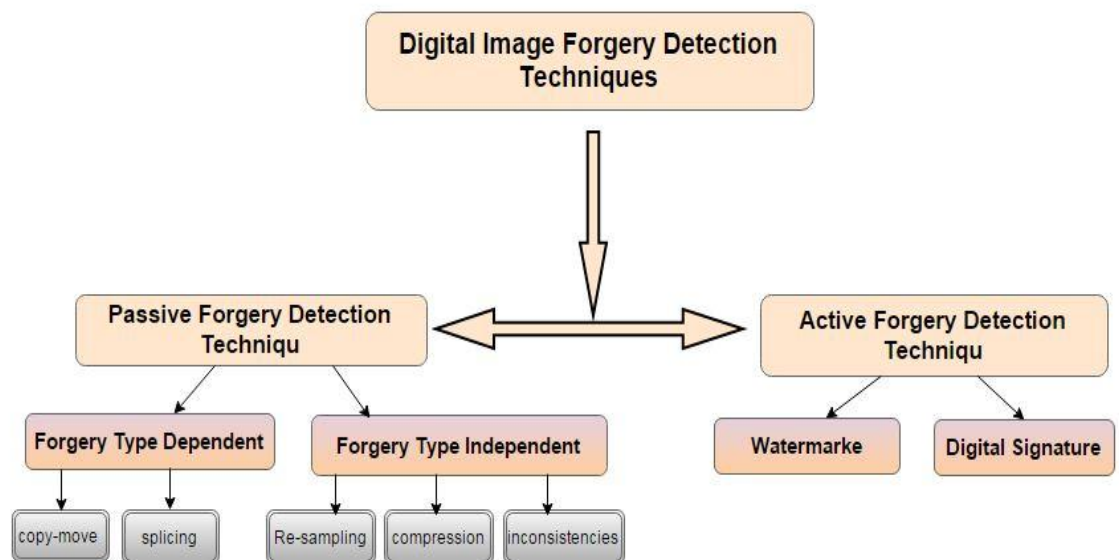


Figure 4 Classifications of Image Forgery Detection Methods

2.1.2.1 DCT-based algorithms

Some of the techniques used to reduce frequency performance [19], such as Discrete Cosine Transform (DCT), effectively find matching areas. They assume that a copy of any processing area, which always has done. This same technique in addition to a slight noise and are lossy compression. Y. Huang, et al. [19] improved this method based on DCT to detect region duplication on an image, where in a area of an image is copied into another area of the same image; therefore the image is considered to be a tampered image by the copy-paste forgery technique. This approach depends on considering the DCT coefficient as features and reducing the feature vector. The algorithm performs better to detect copy-move tampering on an image that is forged by rotation against Popescu's et al. [32] proposed algorithm, which is quicker than Fridrich's, et al [31], because of the dimension reduction. The algorithm has robustness in JPEG Compression, Gaussian blurring and additive white Gaussian noise. The first step divides the image into $B \times B$ fixed size overlapping blocks, followed by the DCT method being applied to each block to extract features. The DCT coefficient generates features which are represented as a row vector in zig-zag order. The matching step tests the blocks' feature vectors. If two vectors are correlative, it means the components of two blocks are very near to each other, which leads to detecting any duplication of two blocks. Finally, the algorithm also outputs the result of duplication visually by mapping the image. Fridrich [31] employs DCT-based features instead of an exhaustive search to identify areas of duplication, which is more effective, in spite of the method being sensitive to variations in the copy given to the additive noise. This work discusses the strength advantage of using the DCT as well as noise, blurring the global and lossy compression. The advantage of utilizing the DCT to describe a feature is decreasing the feature vector size. Moreover, this categories approach presents robustness of several processing procedures, especially JPEG compression operation, additive noise operation and blurring; however, they cannot fight the geometric transformation operations.

2.1.2.2 Algorithms based on PCA

A technique that is used to reduce the size of guarantees [17], [18] includes Principal Component Analysis (PCA). S. Khan's et al. [20] PCA-based approach reduces complexity when using the discrete wavelet transforms (DWT), but still it does not address geometric transformations. In [21], the authors propose their method using a set of properties moment such as PCA and a tree KD- in order to effectively identify areas of duplication. In [33], Popescu has adopted a new method based on features of the PCA, which can add noise tolerance that has been suggested. However, there is little recognition accuracy. The proposed a method in [34] Gharibi et al, here texture considered as a feature. From image blocks the extraction of texture features done by using a Gabor filter, followed by using the PCA to decrease the feature vector size. The approach is robust for JPEG compression operation but not for the other operations of the image processing. The algorithm depends on many initial values and thresholds. The listed studies in this section have relatively low computation costs and small feature vector sizes, but they are not robust against other types of operations.

2.1.2.3 Algorithms based on Log-polar transform

The transform of log polar used to decrease the size of the feature vector in the step of the block-matching phase. The log-polar transform is gained by dividing the Cartesian space coordinates into a radius r and angle u relative to the source of the coordinate system [33]. S. Bravo et al [22] propose a study by means of a 1-D reflection descriptor and rotation-invariant descriptor. The algorithm has these steps: sorting of block, seeking and improvement. Firstly, they divided the image into overlapping blocks of size $(q \times q)$. For every block, a disk of diameter q is determined and four features (f_1, f_2, f_3, f_4) are then evaluated. The four features correspond to the averages of the red, green and blue components, and the fourth one is the entropy of the pixels within the disk. For each block B_i that corresponds to a

descriptor $\bar{v}$ is evaluated by transforming blocks rectangular coordinates into a easy log-polar maps, followed by calculation of the Fourier magnitude. Then calculation of the correlation coefficients among the values of the Fourier magnitude is done. S. Bayram et al. [24] utilized the Fourier-Mellin Transforms (FMT), which is based on mapping the log-polar to symbolize image blocks with attention to rotation and scale in order to permit perfect execution of the algorithm when the copied area are softly rotated and resized. In the step of matching the blocks they applied bloom-filters to minimize the computational complexity. These algorithms are more difficult than the algorithms based on DCT. They have the quality of robustness of intermediate processing; log-polar has invariance to rotation, translation and scaling. However, they are not strong against post processing as the algorithms rely on many thresholds due to requirement of many images depending on number of tests to set the thresholds value to got algorithms best performance.

2.1.2.4 Invariant key-points algorithms

Algorithms in this section is based on non-block, this mean to extract image features the image is not divide into blocks; By using the scale-invariant feature transform (SIFT) method and Speeded Up Robust Features (SURF) method, the features are extracted from the whole image. These approaches are applied to extract special local features and key-point descriptor features. The vectors/descriptor features are constant to rotation, scaling and translation, and are strong to local geometric distortion. I. Amerini et al. [28] suggest a detection method based on a SIFT that first detect and second assessment the geometric transformations which are applied in copy-paste tampering. In three steps the detection is done; SIFT features is extracted in the first step, and match the key-points. Next step, key-point clustering and forgery detection is performed, followed by estimates of the geometric transformations. The results present a high accuracy in the prediction of the different values of the affine transformation, within additive noise and JPEG compression. Likely I. Amerini's algorithm, X. Pan, et al. [35] proposed other detection algorithm,

the geometric transformations estimation in a copy-paste forgery done by using the SIFT-based method. In this approach firstly the tampered image converted into grayscale image. SIFT detection then is invested to detect key-points of an image and calculate features of an image at the found key-points. By the best-bin-first algorithm the found SIFT key-points values are matched with their feature vectors values. Depending on the assumed key-point value matching, by using RANSAC the geometric distortions of the copied areas are estimated. The algorithm shows robustness versus operations of intermediate processing even when they are joint with additive noise as operations of post-processing or JPEG compression. The rating of the affine transform is perfect when duplicated regions are larger. Comparing this category with the other algorithms, it presents robustness against post-processing operations and a wider spectrum of intermediate. However, the algorithm has many disadvantages. It is time consuming and more complicated and. Moreover, the lack of ability to detect the copy-move areas within a highly uniform texture that are its notable key-points value are not retrieved by the SIFT technique or SURF technique. Moreover, the algorithms consist on many thresholds.

2.1.2.5 SVD-based algorithms

A matrix factorization Singular Value Decomposition (SVD) technique is applied on image to extract geometric features and algebraic features. SVD features involve three characteristics: scaling operation, rotation invariance operation and stability [36]. SVD technique has been consumed in various fields, for example pattern recognition, digital signal processing and also data compression. In [37], G. Li et al. present a detection technique that is build on a SVD, Digital Wavelet Transform (DWT) algorithm and strong features for matching the segment. K. Xiaobing et al. [38] study's based on using the theory of reduced-rank approximation to decrease the size of feature vector. This approach's theory supposed only the highest k singular values, and the other values are equal to zero for processing of reducing the matrix. Also this study has a low computational complexity, and it presents high accuracy

and robustness versus JPEG compression operations, retouching blur filtering operations, and Gaussian noise addition operations. The SVD has the ability to detect copy-paste manipulation in forged images attacked by Gaussian blur-filtering, lossy JPEG compression operation and Gaussian white noise contamination. The other advantage of this algorithm is its minimum size of feature vector and comparatively low computation. On the other side, however, the algorithm does not offer the robustness with the other types operations of image processing.

2.1.2.6 Algorithms based on Texture and intensity

Two of the most searched image processing features are texture features and intensity features, by considering texture as a collection of intensity variations that obeys certain repetitive patterns. Texture is an effective feature for image recognition and characterization. Each pixel is influenced by its neighboring pixel values so it is hard to analyze texture values depending on this pixel values. Moreover, for feature extraction descriptors based on illumination invariant intensity are used in distinct points. The substantial advantage of utilizing texture features and intensity features is to minimize the feature vector size.

A. Langille, et al [39] proposed algorithm by seeking for blocks with identical intensity model and that exercises a kd-tree to find the computational complexity. W. Luo et al. [40] suggested a method to detect a copy move forgery by applying seven characteristic features based on intensity. E. Ardizzone et al. [41] proposed algorithms that analyze representation of a bit-plane in image. The first n -bit grayscale image is divided into n different planes and each plane is split into $m \times m$ blocks. Then each block is present as an array of m^2 bits. This array is zero-filled to set its size a multiple of 8. Bits on this array are transforming into characters by applying ASCII code, after that using it's in the matching step. Another study of E. Ardizzone et al. [42] uses criterion texture descriptors to expose duplicated areas using five criterion texture descriptors, Edge histogram, Statistical, H. Tamura et al. [43], R.M. Haralick [44] and Gabor [45]. The first step extracts texture features from

overlapping-blocks then they are saved as vectors. Depending on a vector's component, the blocks are stored. The blocks have a extreme variance over all of the blocks. Then, to find similar blocks, a sorted list is scanned. The proportional errors as the absolute error of a ratio and the lower value of the two vectors are computed. The authors evaluated the robustness of their algorithm for JPEG compression, but they did not examine the robustness of the algorithm versus other processing operations and intermediate. Moreover, the JPEG compressed images edge histogram and the statistical descriptors gave the best results. This category is very close to DCT-based algorithms in terms of advantages such as a minimum size of the feature vector and its credibility. However, these category algorithms do not show robustness versus image processing operations, such as geometric operations.

2.1.2.7 Other algorithms

There are more algorithms that deal with copy-move manipulation detection which have been published recently and which cannot be classified under the previous detection algorithms. B. Dybala et al. [46] suggest a exposition algorithm that depends on the operation of a filtering and searching a nearest neighbor. Wu et al. [47] analyzed the inpainting attack and present using features of connectivity zeros and fuzzy organism to find tampering. Li et al. [48] presented a JPEG images detector that is building on artifacts blocking. In J. Zhao, [49], method is extracted features from image blocks by using the Locally Linear Embedding method (LLE). Here dispute to PCA, LLE can convert high-dimensional data values to low-dimensional data values without wasting information. Khan and Kulkarni [50] exercised the DWT of two levels to decrease the size of the image, then evaluated the correlations of blocks after utilizing FFT to find copy-paste regions. This method is strength against additive noise operation and JPEG compression operation, but it is weak against the other operations of image processing. Muhammed's et al. [51] algorithm decomposes the image by using dyadic undecimated wavelet transforms. They used both LL1 and HH1. The wavelet transform coefficients from every block

are extracting as a feature vector, and the Euclidean distance among each pair of vectors is evaluated. Then sorting the found distances in descending order for HH1 and in ascending order for LL1, and then truncating the results of two lists depending on their threshold values. If a couple of vectors depending on their dimension in both lists show a similar position, this couple of blocks is exposed as a duplicated region. The algorithm is so simple, but it is not solid against the various operation of image processing. Sudha and Kakar [52] used MPEG-7 tools of image signature which are portion of the MPEG-7 standard. The MPEG-7 standard is designed to retrieve fast and strong image and video. By employing the stringent multi-hypothesis matching process, the algorithm can minimize FPR values. However, the algorithm was quite complex and was not robust with a wide domain of intermediate operation and pre-processing operation. M.A. Sekeh's et al. [53] algorithm is based on matching of two-layer blocks to enhance the time intricacy of the matching stage. Two kinds of features are used in this algorithm, low-accuracy feature and high-accuracy feature, and two sequential matching stages. The conclusion shows that the matching technique with two-layer is more functional than the sorting by lexicographic technique in terms of complexity and time.

2.1.3 Video Forgery:

Digital video manipulation has now become an easy task more than before because of the availability of video editing tools. Software allows for easy tampering of a video sequence in a way that is realistic. Moreover, it is possible to manipulate videos in several ways, most of which are common temporal techniques including Frame Drop or Frame Removal, Frame Swapping or Reordering of frame sequences, Frame Addition, Frame Averaging, Duplicating and removing a region from a video sequence scene. Video tampering contains compression by removing temporal frames, temporal redundancy and spatial redundancy. Manipulation occurs by attacking the video contents (i.e. information offered by the video frames), or by attacking the temporal dependency between video frames. Due to the regional

characteristic of the video sequences, video manipulating attacks can be divided into three basic categories: spatial tampering attacks, temporal tampering attacks and spatio-temporal tampering gained from the combination of spatio-temporal tampering attacks [56] as presented in Figure 5.

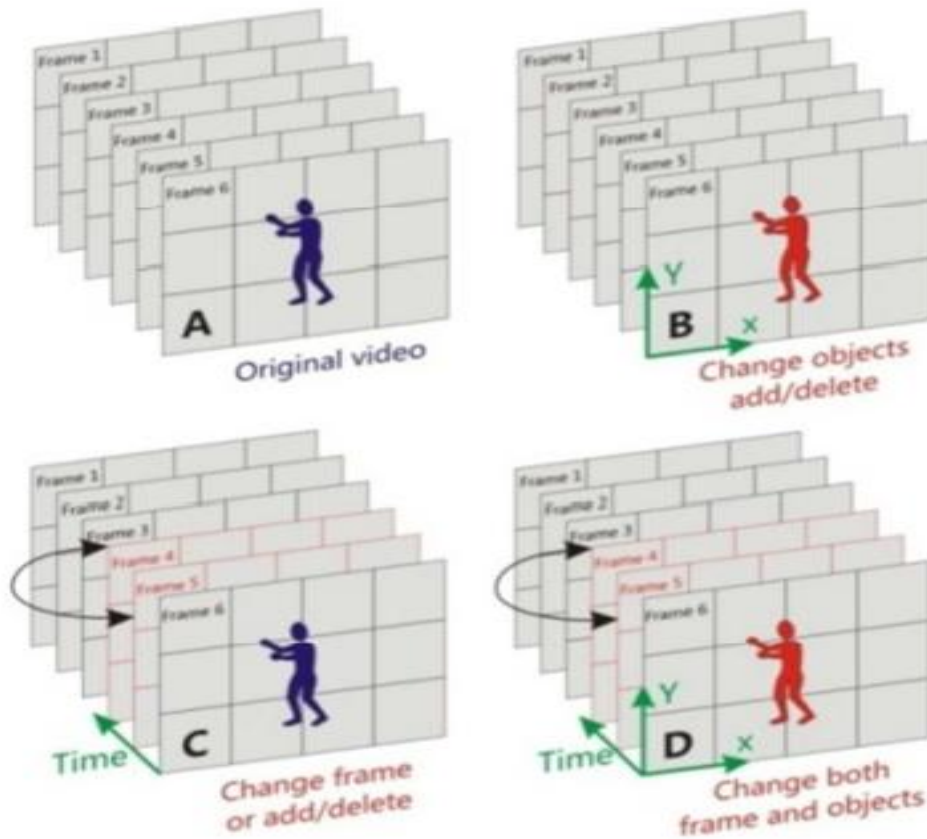


Figure 5 Examples of (A) Original Video Sequences (B) Spatially Tampered Video Sequences (C) Temporally Tampered Video Sequences and (D) Spatio-Temporal Tampered Video Sequences.

Owing to spatially nearby pixels, the forger can manipulate videos spatially (spatial tampering) by manipulating pixel bits within a video frame. Spatial attacks forge the video frame by cutting, copying, pasting, moving and duplicating frames. In

temporal attacks, the video frames are manipulated with respect to time by disturbing the frame sequence, such as frame addition, frame replacement, averaging of frames, frame sequence reordering, and by the removal of video frames. Finally, spatio-temporal tampering is the forgery of video by manipulating pixel bits within a video frame or set of adjacent frames and through distributing the frame sequence [55].

2.1.4 Video Forgery Detection Techniques:

A video forgery detection method aims to find tampering by evaluating the originality of digital video evidence. Video forgery detection techniques basically can be divided into two approaches: active detection approaches, and passive (blind) detection approaches.

The domain of active approach [55] is easily focused on hiding data, so it needs to pre-embed information such as a watermark, or digital signatures into images such as fingerprint, identifying them done by integrity detection of the pre-embedded information. Active detection depends on a watermark or digital signature which can be set only in a few cameras. Most cameras do not have this technology, therefore making an active approach very difficult. On the other hand, the passive video forgery detection approach is more useful for some issues such as video, photo image or audio because good tampering will mislead human notice while statistical and mathematical features of the video or image are altered. This approach aims to extract the internal features of a video for the purpose of detecting forgery without depending on pre-embedded or pre-extracted information. The passive video forgery detection approach has a great scope in multimedia security, pattern recognition and information security [57].

In this thesis, we focus on the passive method techniques because our proposed approach is classified under the passive method techniques.

Passive method techniques can be partition into three general categories; source identification, splicing, and copy-paste forgery. Such approaches are effective in the

detection of most forgery operations such as digital video authenticity using video object detection and double compression video (MPEG or H.246). It is also beneficial to detect video frame of region duplication and frame-based tampering, as shown in Figure 6.

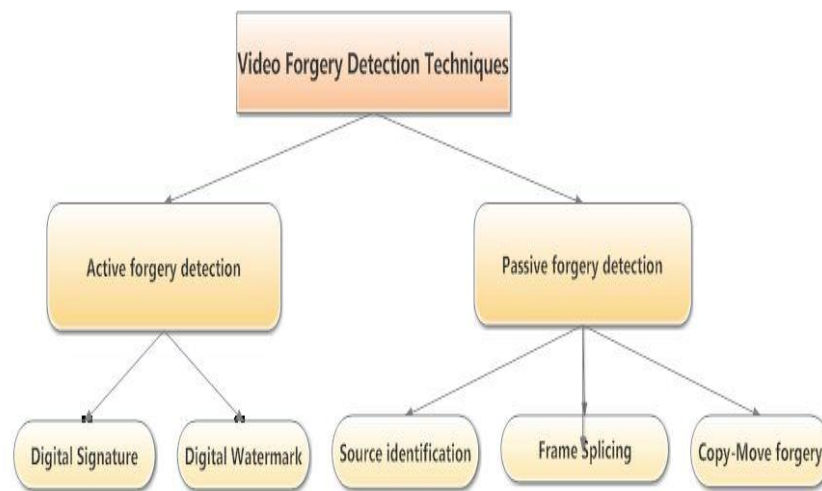


Figure 6 Video Forgery Detection Techniques.

Copy-move manipulation is most common type of video tampering. It focuses on the type of tampering when a section of the frame is cloned and moved into another area in the same frame or another frame, aiming to add or remove an object from the video frame or copying the frame and pasting it into another location of the same video. To detect this type of the forgery, many techniques are introduced most of which are based on the estimation that a copy-move tampering leads to creating a major correlation between the source frames and duplicated frames. Some of these techniques detect the manipulation without localizing the forgery. Therefore, the best methods are those which detect and localize the forgery. The following technique covers the most useful algorithms to detect video forgery with focusing on localization techniques:

2.1.4.1 Algorithms based on HOG Features and Compression Properties:

These algorithms are used in detecting the spatial and temporal copy-move forgery. It is a challenge to detect this kind of forged video, in the terms of size a forged spot may be extremely variable, compression type and rate (I, B or P) or in terms of another changes such as filtering and scaling. A. V. Subramanian and S. Emmanuel's [58] algorithms are builds on the video compression properties and Histogram of Oriented Gradients (HOG) feature matching. HOG evaluates the number of appearances of gradient direction in a domestic spatial area of the image such as cell and size of cell, which may be 4×4 , 6×6 or 8×8 pixels. To extract the HOG features, the gradients of the image are calculated followed by mapping a histogram of orientation at each cell. Finally, a normalized histogram is gained from every cell in a block to product the HOG descriptor block. This algorithm detects spatial forgery; first, it sets the cell size by applying a mechanism for image thresholding. Then it generates the HOG features for every block and for individual block descriptors finds the matches at the same time. Then exposing of the temporal forgery here is presented and depending on the compression properties, the GOP frames are chosen. Then the HOG features are produced block-wise. Finally, to find whether a match exists, they compare these descriptors with the spatially co-located descriptors. The benefit of applying HOG features technique is that they are robust versus various signals processing forgery. However, the authors have not presented the computational efficiency of their algorithm.

2.1.4.2 Algorithm Based on Feature Extraction:

Sheng, YL and Tian, QH [59] used a method based on Tamura texture features. They proposed an algorithm with the help of the vector matrix of the video through video frame extraction. The method calculates the differences between the Tamura texture feature vector and the adjacent vector matrix. By extracting the Tamura texture features of every video frame, the eigenvector matrix of the video is generated

followed by the generated matrix being sorted by dictionary ordering, after which they compute the differences between the feature vector and the adjacent feature vectors. In case the differences are less than the threshold, comparing the distance of the serial number with the threshold, and if the pairs of the serial numbers are higher than the distance threshold, the pairs of serial numbers are sorted to locate the copy-move sequences. This algorithm is efficient to detect and locate frame duplication. Due to the algorithm using frame-level matching, the results show that the precision of this method can reach 99.6%.

2.1.4.3 Algorithm Based on Correlation of Noise:

By using the statistical property of noise residue, C. Hsu, et al. [60] proposed that their approach detects locating forged regions in a video. In this method, for classification of the block-level the value of correlation of noise residuals is extracted as a feature. A Gaussian mixture model (GMM) is modeled by the division of the correlation of provisional noise residual in a manipulated video. They present a two-step to estimate the model parameters. Depending on these parameters, the Bayesian classifier finds the optimal threshold value. The bottom up method is used for defining the tampered/in painted regions in a digital video based on the temporal block-level noise correlation. The first step of the extraction of the noise residual value for every video frame is obtained by subtracts the original video frame from the noise free version of the same frame. The noise free image is gained by using the wavelet de-noising filter. In the next step, firstly every video frame is divided into non overlapping blocks with size of $N \times N$ followed by the illustrated computing from the correlation of the noise residual among the same indexed blocks spatially for two successive frames. The final step is the location of forged blocks by analyzing the statistical properties for block-level noise correlations. In the first section of this step, gaining a coarse distribution of a soft thresholding scheme is exploited. Depending on the coarse classification, a (Gaussian Mixture Model) GMM pattern is utilized to characterize the statistical classification of block-level

temporal noise correlations for the manipulated and non-manipulated regions respectively. By using the EM algorithm, the GMM pattern parameters then are estimated so that they are accordingly deriving the optimal thresholds values is done by using the estimation with maximum-likelihood (ML), and the Bayesian classifier.

2.1.4.4 Algorithm Based on Detection by MPEG Double Compression:

MPEG- x or H.26 x coding standards are usually used to compress digital videos. A forging be operational in an un-compressed domain in order to attain frame insertion and frame deletion. The forged video has to be re-encoded according to factors that include size and format. Thus, digital forgery may be disclosed when double compression occurs. This algorithm invests the properties of video compression, such as periodic properties and blocking artifacts in MPEG1 and MPEG2 videos. Wang and Farid [61] used spatial and temporal effects of double MPEG compression which works on static and temporal artifacts that have been introduced in the video sequence when it undergoes double MPEG compression. Luo et al [62] [63] used temporal patterns of blocking artifacts to determine whether an MPEG video can afford frame insertion or deletion before recompression. They proposed that MPEG compression introduces various block artifacts into unlike frames. Therefore, when a number of frames are removed from an MPEG video file and the file is re-compressed, the block artifacts informed by the previous compression rest and influence the average of block artifact intensity of the re-compressed one, which provides evidence of tampering. Ravi et al [64] proposed a technique to detect forgery in MPEG videos by using the Huber Markov Random Field Model. Their method analyzes the frame's compression noise properties which are extracted from spatial domain. This method is robust to compression artifacts.

2.1.4.5 Algorithm Based on Correlation Matrix:

Wang et al. [65] proposed a method that detects double quantization generated from double MPEG compression in digital video. They computed the differences between the corresponding temporal and spatial domain correlation matrixes. First, a temporal correlation matrix is computed between each frame, but a spatial correlation matrix is calculated for each frame in a given sub-sequence of frames. Highly localized tampering is detected accordingly to a high correlation. The performance of the method is good for detecting frame duplication, but it is not efficient to detect region duplication for small forged regions such as 64×64 . In addition, this technique assumes that the forged region belongs to the same video.

2.1.4.6 Other algorithms:

This section presents the algorithms that deal with video manipulation detection which have been published recently and which cannot be categorized under the previous categories. Shiang Lin et al [66] proposed a course to mulct approach which is based on the concept of a high correlation among the duplicated clips is gained when frame duplication occurs in the processed video. Thus, they used the similarity between two clips as a feature to find these duplicated clips. The method involves three stages: filter clip selection, spatial correlation account and frame duplication classification. First, the histogram difference of two adjacent frames in the RGB color space is adopted to screen duplicated candidates in the temporal domain, followed by estimating the similarity of image content. A block-based approach is used to measure the spatial correlation of each corresponding frame between the requested clip and the candidate frame. Finally, the localization of the duplicated frames occurs via the analysis of spatial and temporal features. This approach is effective for detecting and localizing duplicated clips of high complexity.

Bestagini et al [67] presented a study that exposes video tampering and localizing them in the spatio-temporal domain. This is completely an unsupervised approach. It

detects whether a spatio-temporal region of a sequence was replaced by either a chain of stationary images duplicated in time or a section of the same video taken from a different time period. It deals with image- and video-based attacks separately. For image attacks, the algorithm analyzes the zero motion video remaining difference between pixels in the same spatial position on successive frames. The residual zeros reveal that the image is spliced, after which it searches for frames with a region of zero residual that remains stable in time. Then it finds the maximum 3D bounding volume that includes only zero residual values. In a video-based attack, the algorithm first splits the residual matrix into non-overlapping blocks and then searches for symmetry between each block. This method finds 90% of duplicated block sequences in video tampering; however, with image-based tampering, this method finds 75% of forged pixels. Moreover, the algorithm is time consuming in real time videos.

Gironi et al [68] proposed an algorithm to detect whole frame insertion and deletion in digital videos. The method works even when several codec's are applied to the first and second compression. By focusing on a fixed Group Of Pictures (GOP) encoding, the GOP structure and size are kept static. The encoder encodes a frame by dividing it to macro-blocks (MBs) and codes each MB separately. Here, MBs belonging to I frames are always encoded without referencing other frames. They are mentioned as intra-coded and denote them as I MB, while MBs belonging to predictive-coded frames are encoded by making reference to previous frames or even future frames. These are denoted as P MB. Finally, the encoder has the opportunity to skip a MB. If this MB can be directly copied from a previous frame, these MBs are denoted as S MBs. For detection, the authors measured the Variation Prediction Footprint (VPF). This measure can also be used to perceive the insertion and deletion separately. This method detects tampering and gives good performance even when the second encoding is as robust as the first one. The disadvantage of this technique is that it cannot detect frame tampering when an attacker removes or inserts a whole GOP.

Dong et al [69] proposed a technique, Motion-Compensated Edge Artifact (MCEA), to detect frame-based video tampering. It detects the MCEA difference between neighboring P frames, and by judging whether there are any spikes in the Fourier transform domain after double MPEG compression, a decision is made. The results show that the proposed technique is effective for frame-based tampering, such as adding/deleting frames and GOP structure change. Moreover, it can predict the GOP structure of the original video, but they investigated only P frames for the evaluation without taking into consideration the contribution of B frames.

2.2 Discrete Wavelet Transform (DWT)

The Discrete Wavelet Transform (DWT) in the wavelet domain converts discrete (digital) signals into discrete coefficients. This transform is basically a sampled version of CWT. In state of working with $a1, b1 \in \mathbb{R}$, the $X(a1, b1)$ values are computed through a discrete net:

$$a1 = 2^{-j}, b1 = k \cdot 2^{-j}, j, k \in \mathbb{Z} \quad (2.1)$$

Where this discretization is named dyadic expansion and dyadic location, successively.

The DWT of the genuine signal is acquired by concatenating every coefficient beginning from the last level of decomposition; we will take the variable c_{jk} for the k th the approximation coefficients of level j value, where d_{jk} will have identical meaning for the detail coefficients of level j . A sample of the coefficients gained by Mallat's algorithm is shown in Figure 7 and Figure 8, here simply can be seen in these figures, the detail coefficients proceed to get high values in the noisy areas of the signal [70].

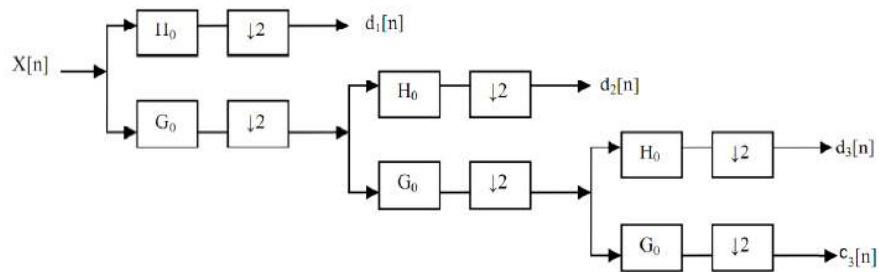


Figure 7 Mallat's algorithm display - 3-level decomposition of a signal. H_0 is an HPF and G_0 is an LPF. $c_j[n]$ indicate the coefficients approximation and $d_j[n]$ indicate the coefficients details

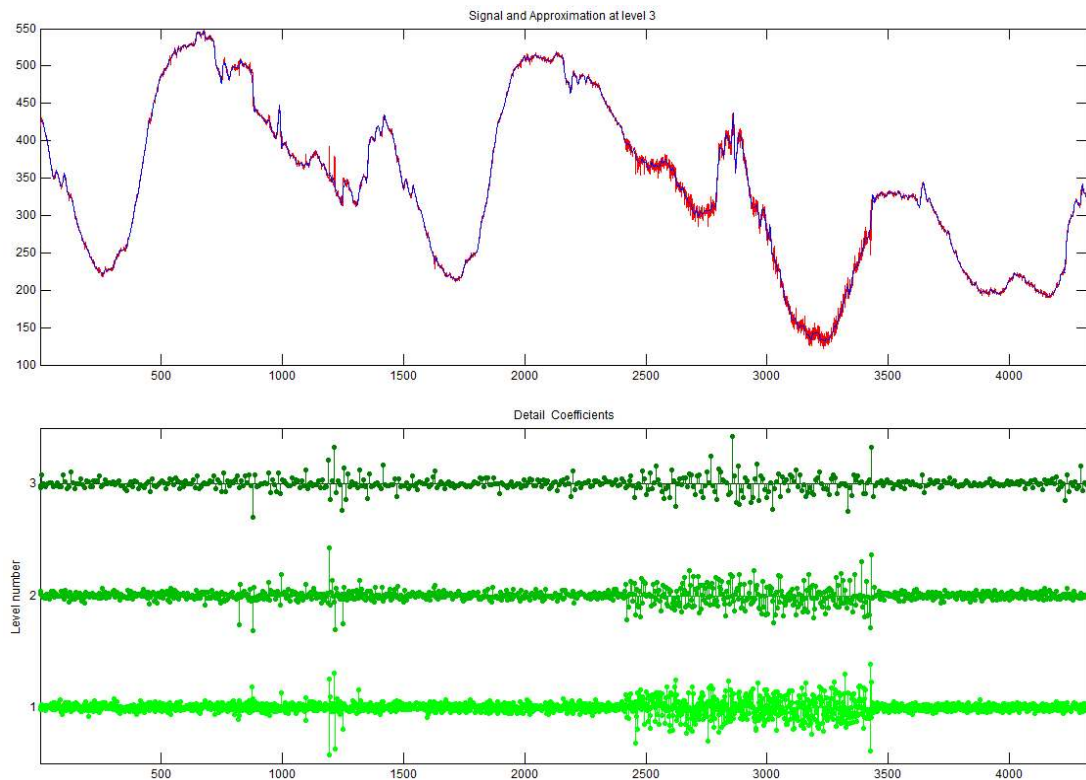


Figure 8 Example of Mallat's model (Symlet the mother wavelet). The signal $X[n]$ and its approximation value (smooth curve) $c_3[n]$ are shown in the upper square, and the three coefficients details are drawn in the lower square.
Formulation of the problem:

A popular presentation of the problem of de-noising is explained down: Suppose that there are $n+1$ noisy pattern of a task f_1 : t

$$y_i = f_1(t_i) + \varepsilon_i, \quad i = 1, \dots, n+1 \quad (2.2)$$

While the noise level σ may be known or unknown and ε_i are iid $N(0; 1)$. Figure 6 present a sample. The aim is to retrieve the underlying action f_1 from the noisy data, $y = (y_1; \dots, y_n)$, with a tiny error when the standard is the mean squared error (MSE). In another words, it is necessary to get a function f which accept:

$$\hat{f}_1 = \min_{\hat{f}} \left\| \hat{f}_1 - f_1 \right\|_2 \quad (2.3)$$

Where $\hat{f}_1 = \hat{f}_1(y)$. It should be obvious in training, the function f_1 is undeclared, and thus the MSE value is commonly estimated. Here the equation (2.2) is not gentle since maybe the noise not added and the relation among the original signal and the spotted signal perhaps stochastic. Still equation (2.2) is a best pattern for several workable situations. It is supposed in what follows, that the t_i are inside the unit interval $[0,1]$ and without loss of generality. Moreover, for simplicity, it is supposed that these model points are equally spaced and that $n = 2^J$ for some of $J \in \mathbb{N}$. These suppositions authorize performing the DWT and the IDWT together using Mallat's speedy algorithm. Figure 9 shows signal f_1 and its noisy version y .

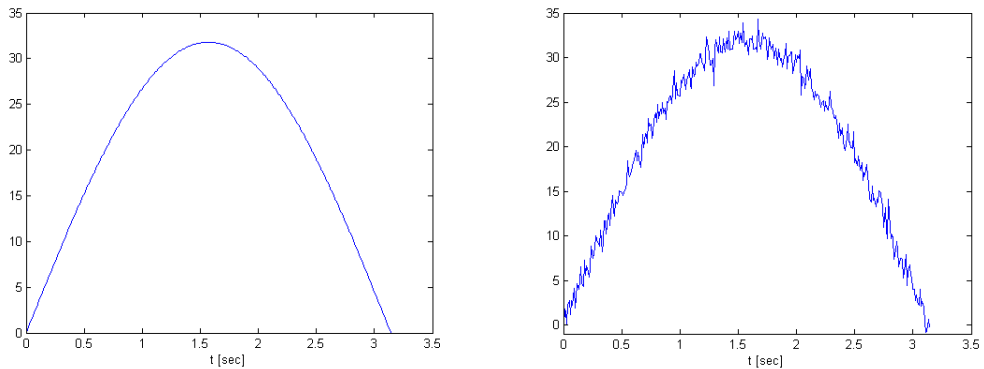


Figure 9 Signal f_1 and its noisy version y

DWT can be used in many fields including mathematics, science, engineering, music, magnetic resonance imaging, fractals, optics, neurophysiology, earthquake-prediction, turbulence, speech discrimination, radar and human vision.

This technique is effective and fast because it

- Supply appropriate information for analysis and synthesis;
- Reduces the time computation sufficiently;
- Is easier to apply;
- Analyzes the signal in various frequency bands at various resolutions;
- Approximation and detail information.

Advantaging from these properties DWT is used in a lot of application the most commons are [72]:

- Compression of digital data, with or without loss.
- De-noising signals.
- In sub-band coding signal and image processing.
- Identifying pure frequencies.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Background

The goal of this thesis is to address passive forgery detection and localization of tampered region in a digital video by using the correlation of noise residual. This works preprocessing is done by extracting correlation of noise residuals values as a feature for classification of the block-level. In the first step, we apply the DWT algorithm to remove the noise from video frames (De-noising) and obtain noise-free frames. In the second step after removing the noise from the frames of each video, frame extraction of the noise residual is done by subtracting the original (noisy version) frame from its noise free version. Then the value of correlation of noise residual between the same spatially indexed blocks for every frame is calculated. After calculating the correlation of the noise residual, the histogram based threshold method checks and finds the correct threshold for the tested video frames. In the final step detection and localization any manipulated blocks by the statistical properties of noise correlations for each block are analyzed then comparing it with the founded threshold value. The proposed method flow chart is shown in Figure 9. This method does not need to pre-collect and pre-train the statistics of noise residual for distinct video cameras such as the noise residual information can be easily extracted from the video to be trusted.

Figure 10 shows the proposed methods flowchart for video forgery detection followed by steps of proposed method explained below.

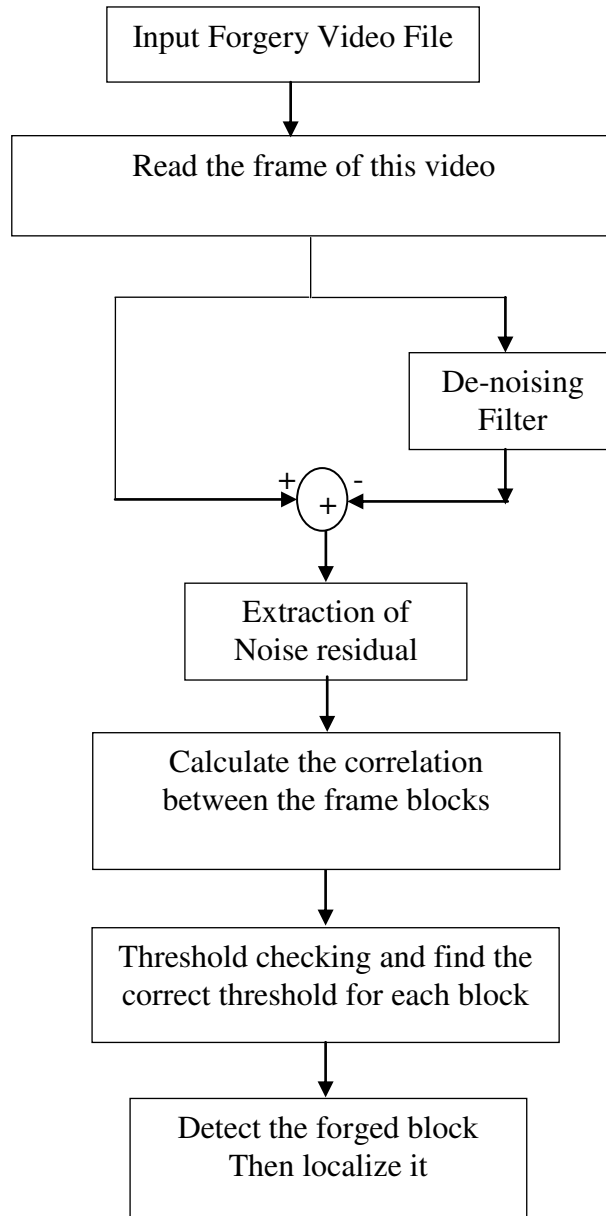


Figure 10 Flowchart of the proposed algorithm.

3.2 Summary of the proposed method

- Input video.
- Read the frames of the input video.
- Apply the DWT algorithm to remove the noise from the video frames (de-noising).
- Extraction of noise residual done by subtracting the original (noisy) forged frame from its de-noised one.
- Block-level noise Correlation values Calculation.
- Check the threshold for every video frame.
- Detect the forged block then localize it.

First the tested forged video file is reading. Then the number of frames that are used in the video file is calculated. For example, for the first video that is used in this thesis, the number of frames is calculated to be about 270 frames.

3.2.1 De-noising Filter

Noising problem is fixed by the de-noising process which covers the task of erasing most of the artifacts because of noise while maintaining the most important image components nearly not harmed to retrieve the coefficients of the original image also it is possible from the noisy perception. In this thesis we adopt the same de-noising filter proposed in [72]. As mentioned in the previous chapter the model of the wavelet coefficients as being locally. Popularize Gaussian was proposed and successfully is used, and due to the property of this model for image de-noising estimating independently and initially underlying variance field by applying a Maximum Likelihood (ML) rule and then using the Minimum Mean Square error (MMSE) estimation procedure. The processes of estimation the variance suppose that the variance area is “locally” soft to allow its safe estimation. By considering the particular case of additive white Gaussian noise (AWGN) in this study, expansions to

more common noise models are possible. The mean-squared error (MSE) chooses as the performance measure.

Here, in Figure 10 $X(k)$ indicate the wavelet coefficients of a “clean” image messed by additive Gaussian noise samples $n(k)$ to yield the observed a noisy images wavelet coefficients $Y(k)$. Figure 11 explicitly display the coefficients of an image $X(k)$ which produced from the multiplication of the outputs of an i.i.d. Gaussian source by samples $\sigma(k)$ from an unknown variance field.

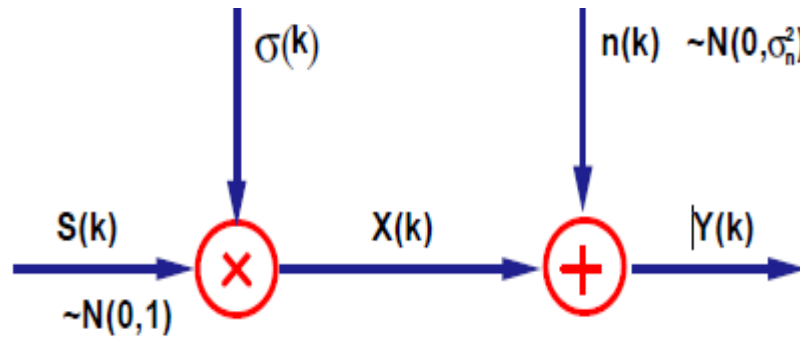


Figure 11 Samples for noisy wavelet coefficients

$X(k)$ refers to the “clean” images wavelet coefficients , each one drawn separately from a Gaussian source with zero mean and variance $\sigma^2(k)$. To produce the observed data $Y(k)$, the $X(k)$ ’s are corrupted by AWGN samples $n(k)$.

The proposed de-noising algorithm is built on the Minimum Mean Square error (MMSE) estimation method [72]. Under the supposition of Gaussianity independence, the optimal predictor (in the MSE sense) for the clean data $X(k)$ is linear and is set by:

$$\hat{X}(k) = \frac{\sigma^2(k)}{\sigma^2(k) + \sigma_n^2} Y(k) \quad (3.1)$$

Where the value of $Y(k)$ is the observed data, $\sigma^2(k)$ is the variance of $X(k)$ and σ_n^2 is the samples of the variance for AWGN [72].

After reading the video frames, for each video frame the de-noising operation is done by applying the Discrete Wavelet Transform algorithm (DWT) to remove the noise.

The de-noising filter algorithm is based on the high-frequency wavelet coefficients that can be shaped as the sum of a static white Gaussian noise and a noise free image. The de-noising process is formed in four steps as mentioned below:

1. Applying four-level wavelet decomposition in a noisy image to gain its wavelet coefficients. After the decomposition is done, in processing step only high-frequency components are used such as the LH $h((i_1, j_1))$, HL $v(i_1, j_1)$ and HH $d((i_1, j_1))$ subbands.
2. Evaluate the local variance of every wavelet coefficient. For every wavelet coefficient, we use a window size of $W1 \times W1$, where $W1 \in \{3, 5, 7, 9\}$. The local variance is computed

$$\text{By: } \sigma^2_w = \max_{(i_1, j_1) \in \Omega} \left(\frac{1}{W^2} \sum_{(i_1, j_1) \in \Omega} c^2(i_1, j_1) - \sigma_0^2 \right) \quad (3.2)$$

Where $\sigma_0 = 5$ $c(i_1, j_1)$ indicate the wavelet coefficient in each sub-band ($h(i_1, j_1)$, $v(i_1, j_1)$, and $d(i_1, j_1)$). Then, we use the minimum value between the local variances of

$$\sigma^2(i_1, j_1) = \min \left(\hat{\sigma}_3^2(i_1, j_1), \hat{\sigma}_5^2(i_1, j_1), \hat{\sigma}_7^2(i_1, j_1), \hat{\sigma}_9^2(i_1, j_1) \right) \quad (3.3)$$

3. For de-noising the Wiener filter, is used as shown in the following profile.

$$c_{den}(i_1, j_1) = c(i_1, j_1) \frac{\hat{\sigma}^2(i_1, j_1)}{\hat{\sigma}^2(i_1, j_1) + \sigma_0^2} \quad (3.4)$$

For each gained wavelet coefficient, we repeat the prior steps till the process converges. Finally, by using the inverse wavelet transform we can achieve the noise free image. The result of the de-noising image is shown in Figure 12.



(a)



(b)

Figure 12 the first frame after implementing the de-noising algorithm (a) original forged image, (b) noise-free image.

3.2.2 Extraction of Noise Residual

For extraction of the noise residual $n(i,j)$ after the noise free image is gained, easily the extraction can be done by subtracting the noisy image Figure 12 (a), from its noise-free version Figure 12 (b). The gain result is shown in Figure 13.

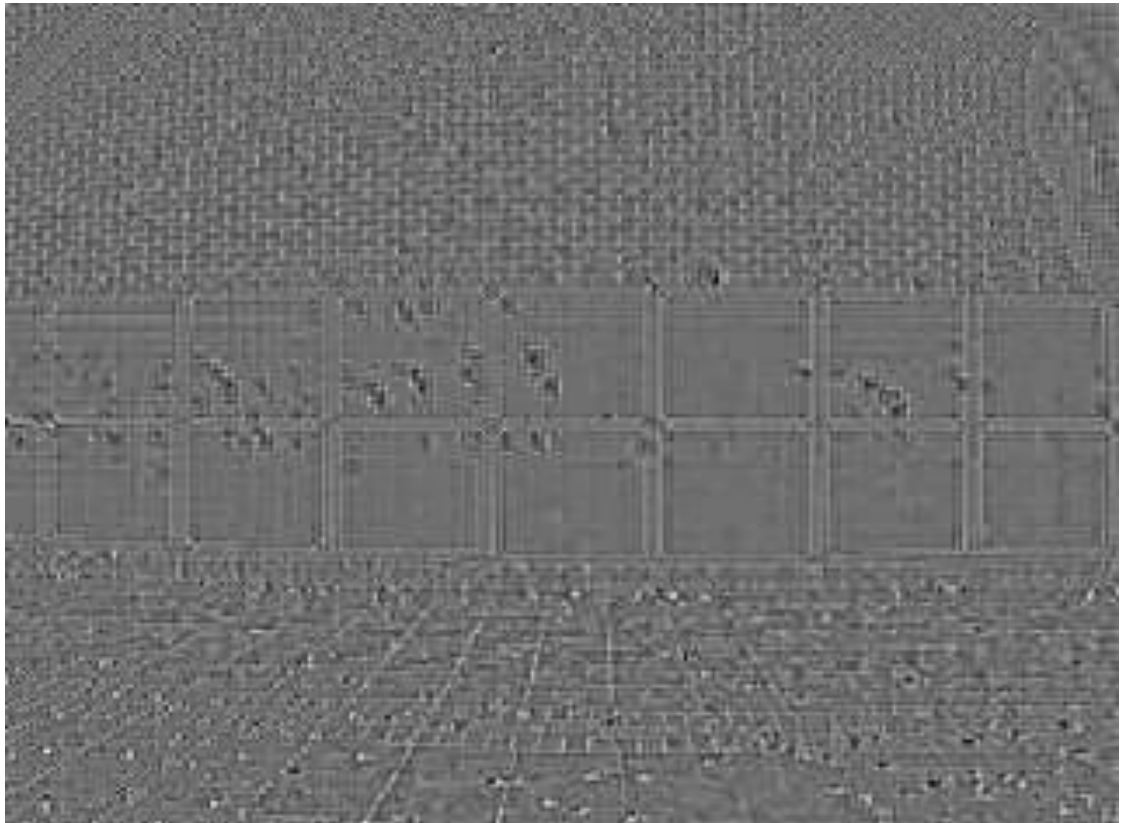


Figure 13 Result of Noise residual extraction

For analyzing the video image, the first channel is considered for all frames. Then each frame will be read and saved in a variable so as to implement the extraction operation on them.

3.2.3 Block-Level Noise Correlation Values Calculation

In this step to calculate the correlation of noise residue, first every video frame is divided into non-overlapping blocks with size $N \times N$. Here in this thesis, for every frame the applied block size is 16×16 . Figure 14 shows the matrix value (pixel intensity) for the first block.

```
[38 79 130 160 172 157 163 167 166 160 164 167 159 158 153 141
68 107 138 153 154 154 159 161 160 156 164 172 170 173 172 161
139 159 161 169 168 175 174 172 170 164 168 174 171 174 175 166
169 168 161 171 172 168 165 169 169 162 164 165 157 160 166 164
174 172 172 177 175 156 162 175 183 178 180 179 166 165 173 174
184 182 182 178 180 163 170 175 163 159 164 166 153 149 154 152
169 166 159 161 169 165 165 162 162 155 162 170 164 164 165 160
163 163 158 173 180 183 183 186 175 162 164 171 168 167 164 154
150 149 153 161 159 152 157 156 155 156 154 151 142 157 168 172
152 154 158 164 161 156 166 171 177 177 173 172 163 177 184 183
181 183 186 186 176 167 177 182 174 172 169 169 162 174 175 169
175 177 180 183 177 169 176 177 174 173 172 173 167 177 175 167
177 177 177 183 184 181 186 182 170 171 172 174 167 176 176 169
180 176 170 174 174 174 179 176 166 168 169 169 161 170 172 167
176 175 170 170 165 166 176 177 178 179 178 175 164 173 176 172
189 191 187 182 170 167 177 182 176 175 170 165 154 162 165 162]
```

Figure 14 Pixel intensity for the first block

The correlation of the noise residual between the same spatially indexed blocks of every frame is then calculated consecutively, as shown in Figure 15.

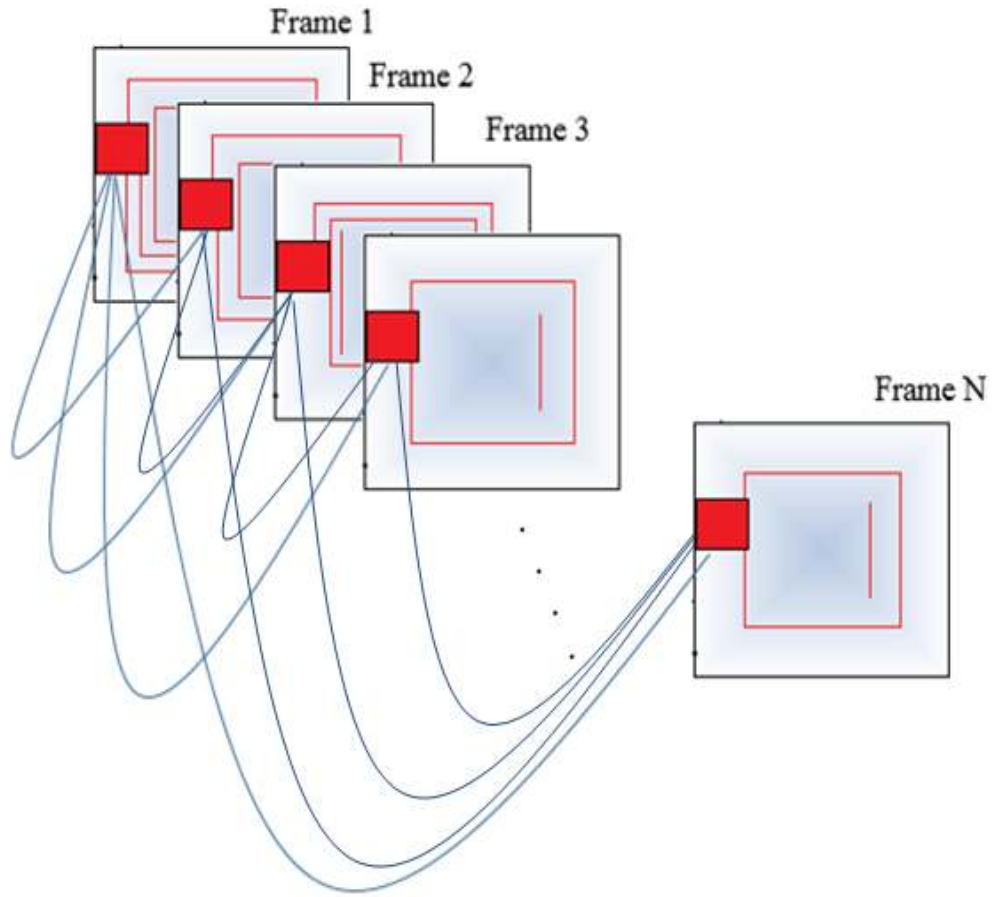


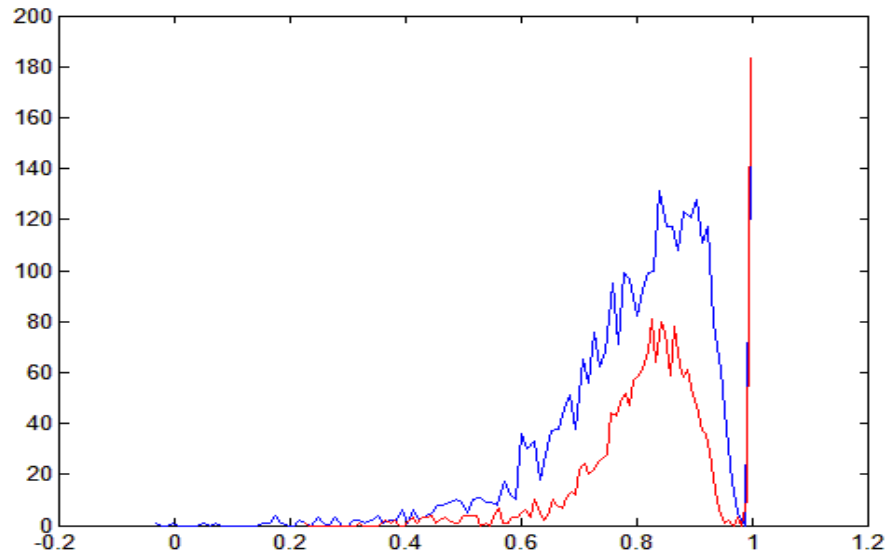
Figure 15 Representation of calculating the correlations of the noise residual between the current frame blocks and previous frame blocks.

The same work will be implemented on each block. For every block, the average is calculated and saved in a variable. Then the correlation between the current frame blocks and every previous frame block in the same position is calculated and saved in variables. This correlation value is the key to detecting forgery as explained in Section 3.2.5.

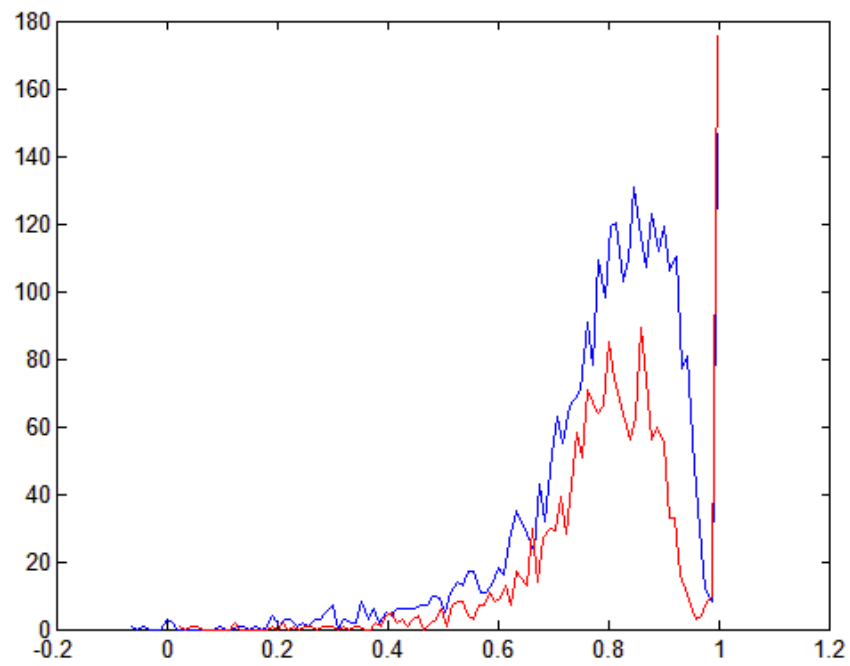
3.2.4 Histogram based thresholding

In this thesis, the correlation threshold value is calculated by histogram thresholding techniques. In general, the threshold should be located in the obvious and deep valley of the histogram. Especially for a well-defined image, its histogram would have a deep valley between two peaks. Therefore, the optimum threshold value can be found in the valley region. One of the most useful methods to find the correct threshold is to find both of the modes (local maxima) and after that find the valley (minimum) between them [73].

When an area in a video is tampered, the value of the correlation noise residual in this area is changed usually it may increased or decreased depending on the used tampering scheme. Here, histogram method calculates the correlation between the current frame noise pattern and the correlation of the previous frames noise pattern for the forged region followed by finding the correct threshold value. Figure 15 present the histogram values for block-level correlation of two successive frames for a tested video 1. The red curves refer to the distributions of non-tampered blocks, whereas the blue curves refer to the forged blocks. This clear from Figure 16 the correlations value in the tampered region are higher, almost relative unity, since in a video with a static background; the content selected for tampering temporally neighboring blocks is commonly the same in order to protect the temporal unity of the forged region.



(a)



(b)

Figure 16 Comparison of noise correlation values between two neighboring video blocks in forged and non-forged regions.

3.2.5 Detecting and Localizing Forgery

The tampering process usually changes the sensor residues statistical properties; we can identify the manipulated regions from the non-manipulated regions by analyzing the statistical properties of the block-level noise correlation. If the correlations value is bigger than the histogram threshold value, and if the threshold value for the average between 20 and 235, then this block is considered to be a forged block.

In this thesis, the threshold value for the average value is selected as 20 and 235 for the minimum and maximum thresholding values respectively.

Occasionally, some blocks are mistakenly detected as forgeries; therefore, to fix this problem, the morphological operator is used to remove this object from the frames.

Figure 17 shows the first block in which a forgery is detected.

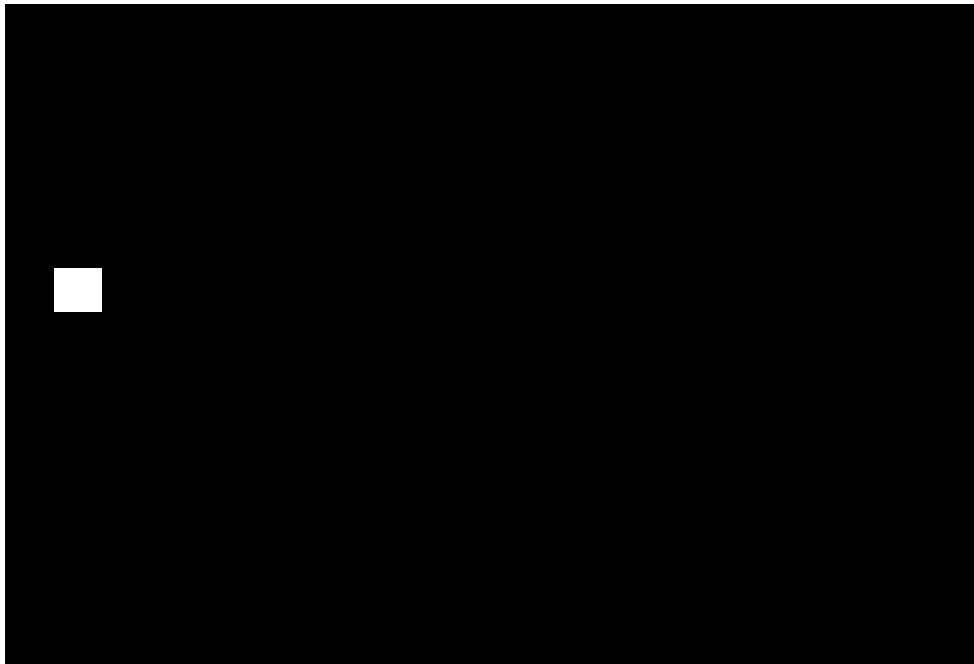


Figure 17 First forgery block

CHAPTER 4

Experimental Results

This chapter describes experiments performed on dataset videos and presents the performance of the proposed approach on the given videos. The localizing step yields different results according to the tested video quality.

4.2 Dataset

Experiments have been conducted on test videos. We collect two datasets, namely original and tampered sets, with each set containing six videos. The database contains all static camera videos with a moving foreground and stationary background. The tampered test videos contain removed objects. Details of both types of test videos are summarized in Table 1 Video sequences in the dataset have been obtained from the Surrey University Library for Forensics Analysis (SULFA) 2012 database. Further details of these videos can be found on the SULFA website [71].

These videos have different resolutions and consist of a sequence of JPEG images, with a frame rate of 25-30 fps for each video. We extract frames from each video sequence with the removed object in different positions.

Video ID	Length	Frame Rate	Video Description	Objects Removed	Source
1	9 seconds	320 × 240	Girl walking	1	SULFA
2	11 seconds	320 × 240	Car moving on road	1	SULFA
3	11 seconds	320 × 240	Man sitting on bench	1	SULFA
4	6 seconds	720 × 480	Man walking on road	1	SULFA
5	6 seconds	720 × 480	Man walking on grass	1	SULFA
6	8 seconds	720 × 480	Man walking on road	1	SULFA

Table 1 Dataset Details

4.3 Experimental Setup

In this study, we run all the experiments on a PC with a 2.50 GHz Intel(R) Core (TM) i5-2450M CPU and 6 GB RAM, using the Windows 7 operating system and the MATLAB 2013, 64-bit software development tool. After implementing the proposed algorithm on the dataset, detection of the forged region is done by comparing the correlation value for each block of video frames with the threshold value for the same video. If the correlation value of this block is greater than the threshold value, it means that this block is forged. Tables 2, 3, 4, 5 show the values of a number of blocks for different groups from test video 1. The forged block is colored green and the results show that the correlation of noise is totally robust versus in fine-quality video; however, it is sensitive to quantization noise. Furthermore, the extraction of the noise residue process is very complex in low quality videos, and the performance of the proposed algorithm is different for each tested video according to the quality of the tested video. Figures 18 to 28 show the visual results of different sampled frames from 6 selected test videos. Each snapshot has 3 groups, two original frames, two forged frames and two forgery detected/localized frames. The obtained result shows a higher performance of forgery detection and localization in high quality videos than the low quality ones.

Frame5										
	0.914986	0.958465	0.954132	0.880791	0.999874	0.999999	0.999999	0.999998	0.999983	0.999999
Frame10										
	0.965258	0.972704	0.950229	0.952109	0.999979	0.999995	1	0.999972	0.999981	0.999952
Frame15										
	0.950689	0.981354	0.977485	0.960889	0.99994	0.999996	0.999997	0.999999	0.999988	0.999998
Frame20										
	0.927602	0.975746	0.963147	0.947111	0.999984	0.999998	0.999999	1	0.99999	0.999999
Frame25										
	0.958165	0.964918	0.975456	0.943667	0.999931	0.999974	0.999991	0.999997	0.999998	0.999998

Table 2 Correlation values for video 1, blocks from frame (5, 10, 15, 20, and 25)

Frame50										
	0.877216	0.950246	0.895452	0.977391	0.981481	0.958021	0.999942	0.999993	0.999998	1
Frame55										
	0.891185	0.978601	0.93494	0.984466	0.976411	0.952232	0.999947	0.999997	0.999994	0.999999
Frame60										
	0.931198	0.981465	0.89368	0.982162	0.977615	0.957662	0.999995	0.999992	0.999998	1
Frame65										
	0.862625	0.983912	0.949974	0.982295	0.985155	0.959596	1	0.999987	0.999999	0.999997
Frame70										
	0.875816	0.980792	0.914614	0.984327	0.985073	0.974501	0.999986	0.999995	0.999999	0.999999

Table 3 Correlation values for video 1, blocks from Frames (50, 55, 60, 65 and 70)

Frame110										
	0.949216	0.93123	0.763451	0.84295	0.999962	0.999996	0.999936	0.999438	0.998572	0.999999
Frame120										
	0.950651	0.956313	0.971555	0.968346	0.977101	0.871377	0.999786	0.999402	0.999736	1
Frame130										
	0.949757	0.94618	0.968513	0.955983	0.984928	0.987772	0.49654	0.980597	0.999842	1
Frame140										
	0.971295	0.949737	0.948183	0.966039	0.956465	0.983561	0.568832	0.870997	0.511495	0.999836
Frame150										
	0.435103	0.942303	0.971085	0.983307	0.976466	0.953654	0.518177	0.920327	0.413327	0.395676

Table 4 Correlation values for video 1, blocks from Frames (110, 120, 130, 140 and 150)

Threshold value for video 1 is 0.9944 comparing it with correlation of blocks in the tables above. The result shows most of the Table 4.2 blocks are forged, meaning the forged region area in this group of frames is larger than other frames. Table 4.4 does not have any forged blocks.

Frame160										
	0.929398	0.914891	0.952538	0.963012	0.942515	0.960396	0.952677	0.949672	0.935624	0.964023
Frame170										
	0.946814	0.961282	0.944739	0.975119	0.968352	0.974932	0.940032	0.950083	0.952182	0.958017
Frame180										
	0.946018	0.959052	0.970808	0.969436	0.950467	0.96129	0.970136	0.950846	0.962962	0.973516
Frame190										
	0.956563	0.933866	0.959452	0.961003	0.956651	0.970023	0.96426	0.957407	0.937505	0.941535
Frame200										
	0.976212	0.921165	0.947242	0.9609	0.969319	0.970246	0.962181	0.924612	0.967439	0.9671

Table 5 Correlation values for video 1, blocks from frame (160, 170, 180, 190 and 200)



(a)



(b)



(c)

Figure 18 Simulation result for video 1, Frames 29 and 94, (a) Source Video frames
(b) Forged video frames (c) Detection and Localization video frames



(a)



(b)



(c)

Figure 19 Simulation result for video 1, Frames 111 and 153, (a) Source Video frames (b) Forged video frames (c) Detection and Localization video frames



(a)



(b)



(c)

Figure 20 Simulation result for video 2, Frames 4 and 9 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)

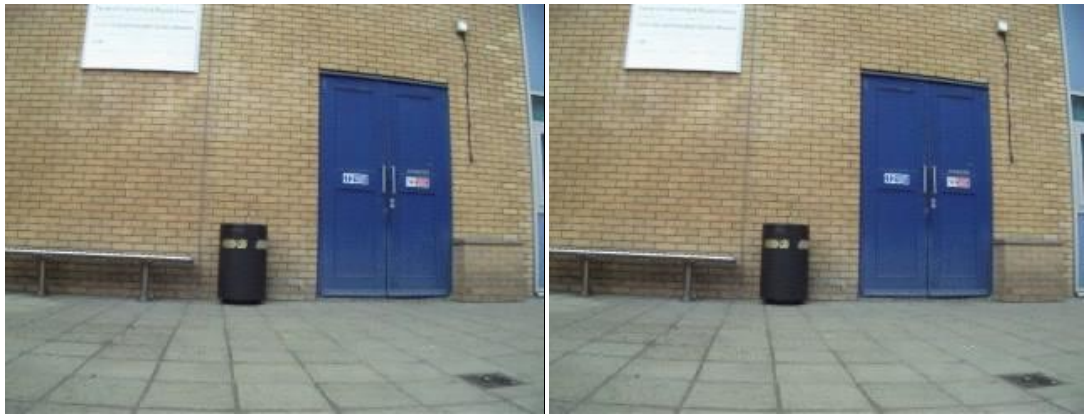


(c)

Figure 21 Simulation result for video 3, Frames 133 and 153 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

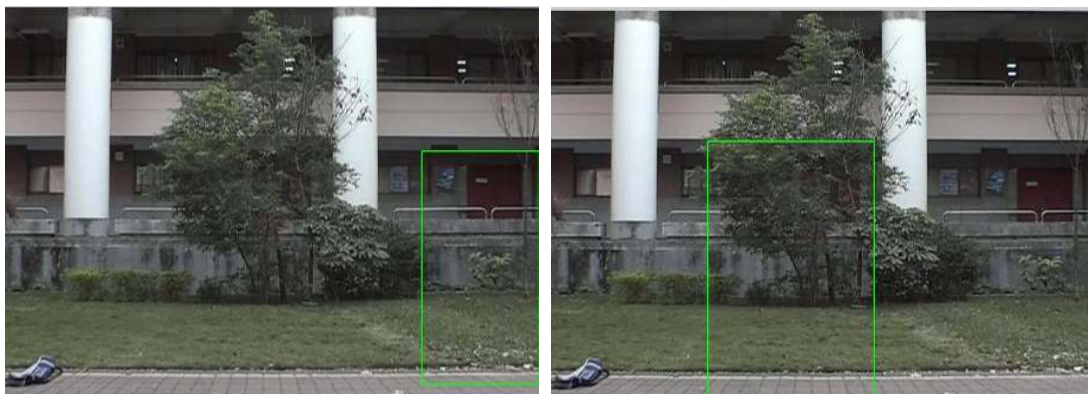
Figure 22 Simulation result for video 3, Frames 182 and 315 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

Figure 23 Simulation result for video 4, Frames 8 and 59 (a) Source Video frames
(b) Forged video frames, (c) Detection and Localization video frames.



(a)

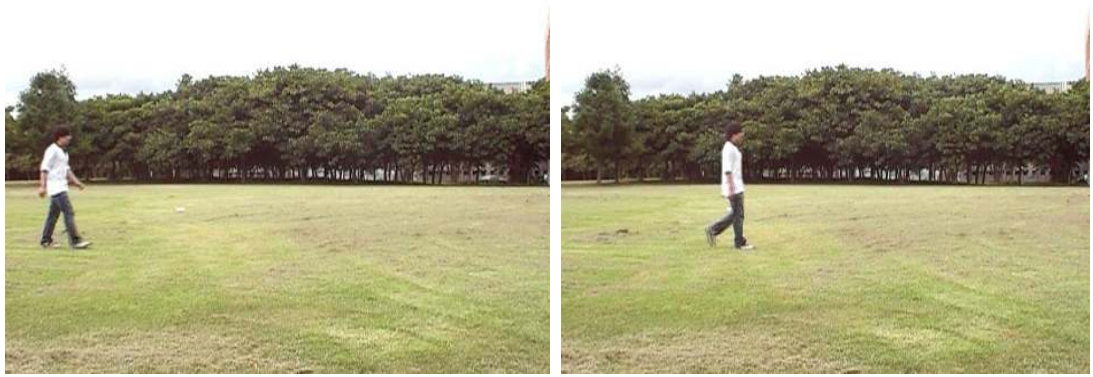


(b)



(c)

Figure 24 Simulation result for video 4, Frames 116 and 136 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

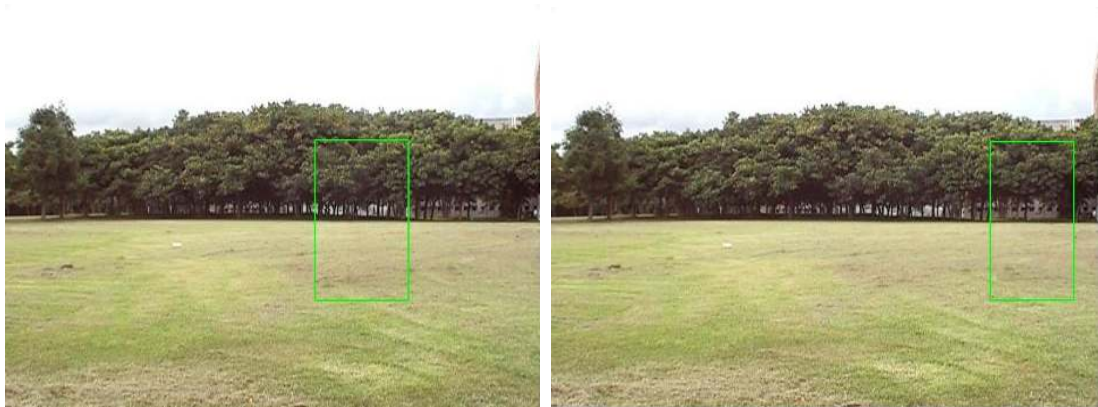
Figure 25 Simulation result for video 5, Frames 8 and 53 (a) Source Video frames
(b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

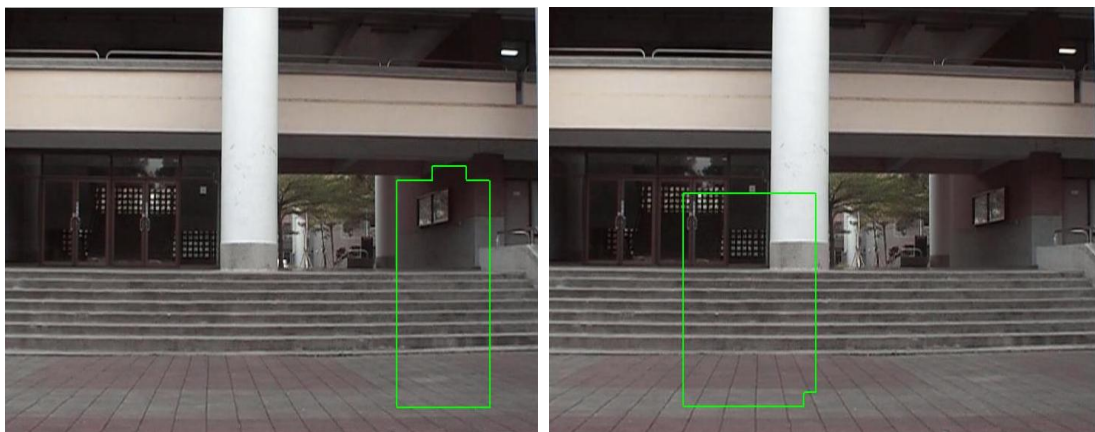
Figure 26 Simulation result for video 5, Frames 118 and 168 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

Figure 27 Simulation result for video 6, Frames 2 and 62 (a) Source Video frames
(b) Forged video frames, (c) Detection and Localization video frames.



(a)



(b)



(c)

Figure 28 Simulation result for video 6, Frames 132 and 198 (a) Source Video frames (b) Forged video frames, (c) Detection and Localization video frames.

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this study, the proposed algorithm aims to detect video manipulation and temporal localization of the tampered part in a video based on correlation of noise residual. First, we begin by presenting an overview of video forensics and we highlight how progression in network technologies, low cost of multimedia devices and digital media editing tools have facilitated production and modification of digital videos with increasing sophistication. Second, we explore the existing forgery detection techniques for image and video and their applied research methodologies. The next part of the study discusses the proposed approach based on correlation of noise residue by applying the DWT algorithm for video forgery detection. Finally, the implemented proposed algorithm on the dataset contains static camera videos with a moving foreground and stationary background which depends on the property of correlation of noise residue. The result shows a high performance of manipulation detection and localization in high quality videos.

5.2 Future Work

In future, we would like to extend this algorithm to videos with camera motion and dynamic background motion. Additionally, we would like to work on detection of further manipulation techniques perform spatial localization in addition to temporal localization in videos.

REFERENCES

1. **Michihiro K., Takahiro O., and Yoichi S., (2010)**, “*Detecting forgery from static-scene video based on inconsistency in noise level functions*”, Information Forensics and Security, IEEE Transactions on, vol. 5.
2. **Joachim J. E. and Bernd G., (2001)**, “*Blind watermarking applied to image authentication*”, In Acoustics, Speech, and Signal Processing. Proceedings. (ICASSP'01). IEEE International Conference on, vol. 3, pp. 1977-1980.
3. **Ramarathnam V., S-M Koon, Mariusz H. J., and Pierre M., (2000)**, “*Robust image hashing*”. In Image Processing, Proceedings. International Conference on, vol. 3, pp 664-666. IEEE.
4. **Ching-Y. L., (2000)**, “*Watermarking and digital signature techniques for multimedia authentication and copyright protection*”. PhD thesis, Columbia University.
5. **Valentina C., Giulia B., Hany F., and E. Roosevelt., (2011)**, “*Active and passive multimedia forensics*”. Citeseer on, vol. 2, pp 8.
6. **Yuping Sh., Fei Lu., Xiaochun C., and Hassan F., (2006)**, “*Video completion for perspective camera under constrained motion*”. In Pattern Recognition, ICPR 2006 18th International Conference on, vol. 3, pp 63-66. IEEE, 2006. 2, 347.
7. **Kedar P., Guillermo S., Marcelo B., (2007)**, “*Video inpainting under constrained camera motion*”. Image Processing, IEEE Transactions on, vol. 16, pp 545-553.

8. **Alasdair N., Andres A., Matthieu F., Yann G., and Patrick P., (2014),** “*Video inpainting of complex scenes*”. *SIAM J. Imaging Sci.*, vol. 7, pp 27.
9. **Chih-Hung L., Chia-Wen L., Chih-Wen S., Yong-Sheng Ch., and Hong-Yuan M. L., (2011),** “*Virtual contour guided video object inpainting using posture mapping and retrieval*”. *Multimedia, IEEE Transactions on*, vol.13 no 2, pp 292-302.
10. **Sreelekshmi D., Gopu D., Shreyas L., Divya D., (2012),** “*Blind detection method for video inpainting forgery*”. *International Journal of Computer Applications* (0975 – 8887) vol. 60 no 11.
11. **Huiping G., Yingjiu L., Anyi L. and Sushil J., (2006),** “*A fragile watermarking scheme for detecting malicious modifications of database relations*”. *Information Sciences*, vol. 176, pp 1350-1378.
12. **Junfeng H., Zhouchen L., Lifeng W. And Xiaoou T., (2006),** “*Detecting doctored jpeg images via dct coefficient analysis*”. *Springer Berlin Heidelberg, Computer Vision-In ECCV* (3), vol. 3953, pp 423-435.
13. **Lyndon K., Shih-Fu Ch., (2008),** “*Internet image archaeology: automatically tracing the manipulation history of photographs on the web*”. In *MM '08: Proceeding of the 16th ACM international conference on Multimedia*, pp 349-358, New York, NY, USA, ACM.
14. **Jessica F., David S., Jan L., (2003),** “*Detection of Copy-move forgery in digital images*”, In *Digital Forensic Research Workshop*.
15. **Hany F., (2009),** “*A survey of image forgery detection*” *IEEE Signal Process. Mag.*, vol. 2, no. 26, pp. 16–25.
16. **Babak M. and Stanislav S., (2010),** “*A bibliography on blind methods for identifying image forgery*”. *Image Commun.*, vol. 25, no. 6, pp. 389–399.

17. **Alin P. And Hany F., (2004)**, “*Exposing digital forgeries by detecting duplicated image regions*”. Dept. Comput. Sci., Dartmouth College, Tech. Rep. TR2004-515.
18. **Hwei-Jen L., Chun-Wei W. and Yang-Ta K., (2009)**, “*Fast copy-move forgery detection*,” . WSEAS Trans. Signal Process., vol. 5, no. 5, pp. 188–197.
19. **Yanping H., Wei L., Wei S. And Dongyang L.,(2011)**, “*Improved DCT-based detection of copy-move forgery in images*”. Forensic Sci. Int., vol. 206, no. 1–3, pp. 178–184.
20. **Saiqa K. and Arun K., (2010)**, “*Reduced time complexity for detection of copy-move forgery using discrete wavelet transform*” .Int. J. Computer Applic. IJCA, vol. 6, no. 7, pp. 31–36.
21. **Babak M. and Stanislav S., (2007)**, “*Detection of copy-move forgery using a method based on blur moment invariants*”. Forensic Sci. Int., vol. 171, no. 27–3, pp. 180–189.
22. **Sergio B. S. and Asoke K. N., (2009)**, “*Passive forensic method for detecting duplicated regions affected by reflection, rotation and scaling*”. In IEEE Signal Processing Conf., pp. 824–828.
23. **Seung-Jin R., Min-Jeong L. and Heung-Kyu L., (2010)**, “*Detection of copy-rotate-move forgery using zernike moments*”. In Proc. Int. Workshop Information Hiding, pp. 51–65, Springer.
24. **Sevinc B., Husrev S., and Nasir M., (2009)**, “*An efficient and robust method for detecting copy-move forgery*” . In Proc. IEEE Int. Conf. Acoustics, Speech Signal Processing, pp. 1053–1056.

25. **Weihai L. And Nenghai Y., (2010)**, “*Rotation robust detection of copy-move forgery*”. In Proc. IEEE Int. Conf. Image Processing, pp 2113–2116.
26. **Hiling H., Weiqiang G., and Yu. Z.,(2008)**, “*Detection of copy-move forgery in digital images using SIFT algorithm*”. In Proc. IEEE Pacific-Asia Workshop Computational Intelligence Industrial Applic., pp 272–276.
27. **Xunyu P. and Siwei L., (2010)**, “*Region duplication detection using image feature matching*”. IEEE Trans. Inform. Forensics Security, vol. 5, no. 4, pp. 857–867.
28. **Irene A., Lamberto B., Roberto C., Alberto D. B., and Giuseppe S., (2011)**, “*A SIFT-based forensic method for copy-move attack detection and transformation recovery*”. IEEE Trans. Inform. Forensics Security, vol. 6, no. 3, pp. 1099–1110.
29. **Hany F., (2006)**, “ *Exposing digital forgeries in scientific images* ”. presented MM&Sec, proceedings of the 8th workshop on Multimedia and security , pp 29-36 ACM New York, NY, USA.
30. **Weiqi L., Jiwu H. and Guoping Q., (2006)**, “ *Robust Detection of Region-Duplication Forgery in Digital Image*”. In Pattern Recognition, ICPR , 18th International IEEE Conference on, pp. 746-749.
31. **Jessica F, David S., and Jan L., (2003)**, “ *Detection of copy-move forgery in digital images* ”. In: Digital Forensic Research Workshop, Cleveland, OH, pp. 19–23.
32. **Chee S. W. , Dong K. P. , and Soo-Jun P., (2002)**, “ *Efficient use of MPEG-7 edge histogram descriptor* ”. ETRI Journal, vol. 24 ,no.1, pp. 23–30.
33. **Osamah M. Al-Q. , Bee Ee Kh., (2013)**, “ *Passive detection of copy-move forgery in digital images: State-of-the-art* ”. Forensic Science International, vol. 231, pp. 284–295.

34. **Fereshteh Gh., Javad R. , Fardin A. , Bahram Z. A. And Javad A., (2011)**
, “ *Robust detection of copy-move forgery using texture features* ”. In 19th
Iranian Conference on Electrical Engineering, ICEE, pp.1-4.

35. **Xunyu P. and Siwei L., (2010),** “ *Detecting image region duplication using
SIFT features* ”. In: Acoustics Speech and Signal Processing (ICASSP), IEEE
International Conference, vol. 978 no.1, pp. 1706–1709.

36. **Klema, Virginia C., and Alan J. Laub., (1980),** “ *Singular value
decomposition: its computation and some applications*”. Automatic Control,
IEEE Transactions on vol. 25, no.2, pp.164–176.

37. **Li G., Qiong W., Dan T., and Shaojie S., (2007),** “*A sorted neighborhood
approach for detecting duplicated regions in image forgeries based on DWT
and SVD*” . In Multimedia and Expo, IEEE International Conference on, pp.
1750-1753.

38. **Kang X., Wei S., (2008),** “*Identifying tampered regions using singular value
decomposition in digital image forensics*”. InComputer Science and Software
Engineering, International Conference ,Vol. 3, pp. 926-930. IEEE.

39. **Langille A., Gong M., (2006),** “*An efficient match-based duplication
detection algorithm* ”. InComputer and Robot Vision. The 3rd Canadian
Conference on (pp. 64-64). IEEE.

40. **Luo W., Huang J. and Qiu, G., (2006),** “*Robust detection of region-
duplication forgery in digital image* ”. In Pattern Recognition, ICPR 2006. 18th
International Conference on vol. 4, pp. 746-749. IEEE.

41. **Ardizzone E., Mazzola G., (2009),** “ *Detection of duplicated regions in
tampered digital images by bit-plane analysis*”. In Image Analysis and
Processing–ICIAP (pp. 893-901). Springer Berlin Heidelberg.

42. **Ardizzone E., Bruno A. And Mazzola G., (2010),** “*Copy-move forgery detection via texture description* ”. In Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence ,pp. 59-64. ACM.

43. **Tamura H., Mori S. and Yamawaki T., (1978),** “*Textural features corresponding to visual perception* ”. Systems, Man and Cybernetics, IEEE Transactions, vol.8, np. 6, pp.460-73.

44. **Haralick R.M., (1979).** “*Statistical and structural approaches to texture*”. *Proceedings of the IEEE*, vol. 67, no. 5, pp.786-804.

45. **Jain A.K. and Farrokhnia F., (1990),** “*Unsupervised texture segmentation using Gabor filters*”. In Systems, Man and Cybernetics, Conference Proceedings., IEEE International Conference on , pp. 14-19. IEEE.

46. **B. Dybala, B. Jennings, D. Letscher,** Detecting filtered cloning in digital images, in: Proceedings of the 9th Workshop on Multimedia & Security, ACM Dallas, Texas, USA, 2007.

47. **Wu Q., Sun S.J., Zhu W. and Tu, D., (2008),** “*Detection of digital doctoring in exemplar-based inpainted images*”. In Machine Learning and Cybernetics, International Conference on vol. 3, pp. 1222-1226. IEEE.

48. **Li W., Yuan Y. and Yu N., (2009),** “*Passive detection of doctored JPEG image via block artifact grid extraction*”. *Signal Processing*, vol. 89, no. 9, pp.1821-1829.

49. **Junhong Z., (2010),** “*Detection of copy-move forgery based on one improved LLE method*”. In Advanced Computer Control (ICACC), 2010 2nd International Conference on vol. 4, pp. 547-550. IEEE.

50. **Khan S. and Kulkarni A., (2010),** “*Reduced time complexity for detection of copy-move forgery using discrete wavelet transform*”. *International Journal of Computer Applications IJCA*, vol. 6, no. 7, pp.31-36.

51. **Ghulam M., Muhammad H., Khalid Kh., George B., (2011),** *“Blind copy move image forgery detection using dyadic undecimated wavelet transform”*. On Digital Signal Processing DSP, in: 17th International Conference, Corfu, pp: 1-6.
52. **Bober, M. and Brasnett, P., (2009),** *“MPEG-7 visual signature tools”*. In Multimedia and Expo, ICME. IEEE International Conference on (pp. 1540-1543).
53. **Sekeh, M.A., Maarof, M.A., Rohani, M.F. and Motiei, M., (2011),** *“Efficient image block matching algorithm with two layer feature extraction”*. In Information Technology in Asia (CITA 11), 7th International Conference on (pp. 1-5). IEEE.
54. **Kim. S. P. Ann. B. L. and David. M., (2003),** *“The nonlinear statistics of highcontrast patches in natural images”*. International Journal of Computer Vision, 54(1-3), pp.83-103.
55. **Gavade J. D. And Chougule S.R., (2015),** *“Review of Techniques of Digital Video Forgery Detection”*, Advances in Computer Science and Information Technology (ACSIT) Print ISSN: 2393-9907; Online ISSN: 2393-9915; Volume 2, Number 3; pp. 233-236 .
56. **Saurabh U., Sanjay K. S., (2012),** *“Video Authentication: Issues and challenges”*. in IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 3, ISSN (Online): 1694-0814.
57. **Wahab A.W.A., Bagiwa M.A., Idris M.Y.I., Khan S., Razak Z. and Ariffin M.R.K., (2014),** *“Passive video forgery detection techniques: a survey”*. In Information Assurance and Security (IAS), on 10th International Conference IEEE, pp. 29-34.
58. **Subramanyam, A. V. and Sabu E., (2012),** *“Video forgery detection using HOG features and compression properties”*. Multimedia Signal Processing (MMSP), IEEE 14th International Workshop on pp. 89 - 94.

59. **Sheng, YL. And Tian, Q H., (2013),** “*Video Copy-Move Forgery Detection and Localization Based on Tamura Texture Features*”. In International Congress on Image and Signal processing (CISP 2013), pp. 864-868.

60. **Hsu C.C., Hung T.Y., Lin C.W. and Hsu C.T., (2008),** “*Video forgery detection using correlation of noise residue*”. In Multimedia Signal Processing, IEEE 10th Workshop on pp. 170-174.

61. **Weihong W. and Hany F., (2007),** “*Exposing digital forgeries in interlaced and deinterlaced video*”. Information Forensics and Security, IEEE Transactions on, vol:2, pp. 438-449.

62. **Luo W., Min Wu., and Jiwu H. (2008),** “*MPEG recompression detection based on block artifacts*”. In Electronic Imaging, pp. 68190X-68190X. International Society for Optics and Photonics.

63. **Luo W., Zhenhua Qu., Jiwu H. and Guoping Q.,(2007),** “*A novel method for detecting cropped and recompressed image block*”. In Acoustics, Speech and Signal Processing, ICASSP, IEEE Inter-national Conference on, vol:2, pp. II217.

64. **Ravi H., Subramanyam A.V., Gupta G. and Kumar, B.A., (2014),** “*Compression noise based video forgery detection*”. In Image Processing (ICIP), IEEE International Conference on pp. 5352-5356. IEEE.

65. **Wang W. and Farid H., (2007),** “*Exposing digital forgeries in video by detecting duplication*”. In Proceedings of the 9th workshop on Multimedia & security, pp. 35-42. ACM.

66. **Lin G.S., Chang J.F. and Chuang C.H., (2011),** “*Detecting frame duplication based on spatial and temporal analyses*”. In Computer Science & Education (ICCSE), 6th International Conference on pp. 1396-1399. IEEE.

67. **Bestagini P., Milani S., Tagliasacchi M. and Tubaro S., (2013)**, “*Local tampering detection in video sequences*”. In Multimedia Signal Processing (MMSP), IEEE 15th International Workshop on pp. 488-493. IEEE.
68. **Gironi A., Fontani M., Bianchi T., Piva, A. and Barni M., (2014)**, “*A video forensic technique for detecting frame deletion and insertion*”. In Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on pp. 6226-6230.
69. **Dong Q., Yang G. and Zhu N., (2012)**, “*A MCEA based passive forensics scheme for detecting frame-based video tampering*”. Digital Investigation, vol: 9, no.2, pp.151-159.
70. **Walker J.S. and Chen Y.J., (2000)**, “*Image denoising using tree-based wavelet subband correlations and shrinkage*”. Optical Engineering, vol:39, no.11, pp.2900-2908.
71. **Qadir G., Yahaya S. and Ho A.T., (2012)**, “*Surrey university library for forensic analysis (SULFA) of video content*”. In Image Processing (IPR), IET Conference on pp. 1-6, London, <http://sulfa.cs.surrey.ac.uk>.
72. **Mihcak M.K., Kozintsev I. and Ramchandran K., (1999)**, “*Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising*”. In Acoustics, Speech, and Signal Processing, Proceedings, IEEE International Conference on vol: 6, pp. 3253-3256.
73. **Arifin A.Z. and Asano A., (2004)**, “*Image thresholding by histogram segmentation using discriminate analysis*”. In Proceedings of Indonesia–Japan Joint Scientific Symposium, pp. 169-174.

APPENDICES A

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: HAZZA, Nidhal

Date and Place of Birth: 16 August 1977, Kirkuk

Marital Status: Single

Phone: +90 507 071 6818 / +964 7701326900

Email: nidal772004@yahoo.com / nidhalbayatli@gmail.com



EDUCATION

Degree	Institution	Year of Graduation
M.Sc.	Çankaya University Mathematics and Computer science	2016
B.Sc.	Baghdad University	1999
High School	Al-Huda High School	1995

WORK EXPERIENCE

Year	Place	Enrollment
1999-2004	Kirkuk Technical College	Lecturer
2004-Present	Kirkuk University College of Science	Programmer

FOREIGN LANGUAGES

Arabic, Turkmen, English, Turkish.

HOBBIES

Reading, Shopping, Travel **and** Cooking.