UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE INFORMÁTICA

TESIS DOCTORAL

Técnicas Pasivas de Detección y Localización de Manipulaciones en Imágenes Digitales

Passive Techniques for Detecting and Locating Manipulations in Digital Images

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Esteban Alejandro Armas Vega

Directores

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Madrid

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

TESIS DOCTORAL

Técnicas Pasivas de Detección y Localización de Manipulaciones en Imágenes Digitales
Passive Techniques for Detecting and Locating Manipulations in Digital Images

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Esteban Alejandro Armas Vega

DIRECTOR

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Técnicas Pasivas de Detección y Localización de Manipulaciones en Imágenes Digitales

– –

Passive Techniques for Detecting and Locating Manipulations in Digital Images

Thesis by

Esteban Alejandro Armas Vega

In Partial Fulfillment of the Requirements for the Degree of
Doctor por la Universidad Complutense de Madrid en el
Programa de Doctorado en Ingeniería Informática

Advisors

Luis Javier García Villalba

Ana Lucila Sandoval Orozco

Facultad de Informática
Universidad Complutense de Madrid

Madrid, July 2020

Passive Techniques for Detecting and Locating Manipulations in Digital Images

Thesis by

Esteban Alejandro Armas Vega

In Partial Fulfillment of the Requirements for the Degree of
Doctor por la Universidad Complutense de Madrid en el
Programa de Doctorado en Ingeniería Informática

Advisors

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Facultad de Informática
Universidad Complutense de Madrid

Madrid, July 2020

Dissertation submitted by Esteban Alejandro Armas Vega to the *Faculty of Computer Science and Engineering* of the *Universidad Complutense de Madrid* in Partial Fulfillment of the Requirements for the Degree of *Doctor por la Universidad Complutense de Madrid en el Programa de Doctorado en Ingeniería Informática.*

(Submitted *July 07*, 2020)

Title:

Passive Techniques for Detecting and Locating Manipulations in Digital Images

PhD Candidate:

Esteban Alejandro Armas Vega (esarmas@ucm.es)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid
28040 Madrid, Spain

Advisors:

Luis Javier García Villalba (javiergv@fdi.ucm.es)
Ana Lucila Sandoval Orozco (asandoval@fdi.ucm.es)

This work has been done within the Group of Analysis, Security and Systems (GASS, http://gass.ucm.es/), Research Group 910623 from the Universidad Complutense de Madrid (UCM) as part of the activities of the research project funded by the European Commission Horizon 2020 Programme under Grant Agreement number H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially Motivated Malware). This work has also received funding from THEIA (Techniques for Integrity and Authentication of Multimedia Files of Mobile Devices) UCM project (FEI-EU-19-04).

Acknowledgments

To God and my family. To my friends and thesis advisors Ana and Javier, thank you for trusted me and have known how to give me their help and guidance during all the stages in the development of this work. To them, the greatest and most profound of my thanks.

This work has been done within the Group of Analysis, Security and Systems (GASS, http://gass.ucm.es/), Research Group 910623 from the Universidad Complutense de Madrid (UCM) as part of the activities of the research project funded by the European Commission Horizon 2020 Programme under Grant Agreement number H2020-FCT-2015/700326-RAMSES (Internet Forensic Platform for Tracking the Money Flow of Financially Motivated Malware). This work has also received funding from THEIA (Techniques for Integrity and Authentication of Multimedia Files of Mobile Devices) UCM project (FEI-EU-19-04).

Contents

List of Figures

List of Tables

List of Algorithms

Resumen

El número de cámaras digitales integradas en dispositivos móviles así como su uso en la vida cotidiana está en continuo crecimiento. Diariamente gran cantidad de imágenes digitales, generadas o no por este tipo de dispositivos, circulan en Internet o son utilizadas como evidencias o pruebas en procesos judiciales. Como consecuencia, el análisis forense de imágenes digitales cobra importancia en multitud de situaciones de la vida real.

El análisis forense de imágenes digitales se divide en dos grandes ramas: autenticidad de imágenes digitales e identificación de la fuente de adquisición de una imagen. La primera trata de discernir si una imagen ha sufrido algún procesamiento posterior al de su creación, es decir, que no haya sido manipulada. La segunda pretende identificar el dispositivo que generó la imagen digital. La verificación de la autenticidad de imágenes digitales se puede llevar a cabo mediante técnicas activas y técnicas pasivas de análisis forense. Las técnicas activas se fundamentan en que las imágenes digitales cuentan con "marcas" presentes desde su creación, de forma que cualquier tipo de alteración que se realice con posterioridad a su generación, modificará las mismas, y, por tanto, permitirán detectar si ha existido un posible post-proceso o manipulación. Por otro lado, las técnicas pasivas realizan el análisis de la autenticidad extrayendo características de la imagen. Cualquier operación de edición sobre una imagen dada dejará "huellas" que serán utilizadas para comprobar la autenticidad de la misma. Las técnicas pasivas son las más utilizadas puesto que no todos los dispositivos tienen la capacidad de dejar "marcas" en la imagen que permitan validar su autenticidad.

En este trabajo se proponen cinco técnicas pasivas:

La primera combina el análisis de patrones locales de textura con el uso de las características intrínsecas de la compresión y el formato de cada uno de los bloques de una imagen para extraer un grupo de singularidades con los que formar un vector que, posteriormente, será utilizado en una máquina de soporte vectorial para crear un modelo que permitirá verificar la autenticidad de la imagen, identificando distintos tipos de manipulaciones como Copiar-Mover o Empalme.

La segunda estima, sin un entrenamiento previo, el patrón de interpolación utilizado por el sensor de la cámara que generó la imagen, siendo capaz de localizar las inconsistencias presentes en la misma que surgen en un proceso de edición posterior como manipulaciones de tipo Copiar-Mover o Empalme.

La tercera, basada en la compresión JPEG, analiza de nivel de error que resalta los píxeles que tienen un nivel de compresión distinto al resto de la imagen y los marca como contenido no original de la misma, detectando manipulaciones de tipo Empalme.

La cuarta, que se fundamenta en los algoritmos de interpolación cromática, divide la imagen en bloques superpuestos y estima el patrón de interpolación y el error cuadrático medio, detectando también manipulaciones de tipo Empalme.

La quinta y última utiliza la transformada discreta del coseno. A través de estos coeficientes se obtienen las características que forman los vectores de transferencia que se utiliza. Estos vectores se agrupan y utilizando un umbral de tolerancia, se puede identificar las áreas duplicadas en una imagen.

Todas estas técnicas han sido evaluadas utilizando distintos datasets de imágenes, públicos y privados, demostrándose su validez al suponer un avance en el estado del arte de las técnicas pasivas de análisis forense de imágenes.

Finalmente, señalar que estas técnicas han sido implementadas en una herramienta de análisis forense, desarrollada dentro de un proyecto del Programa Marco de Investigación e Innovación de la Unión Europea Horizonte 2020, herramienta que ha sido además validada por Policías de diversos países europeos en un entorno real.

Palabras clave: Análisis Forense, Autenticidad, Características, Copiar-Mover, Crominancia, Detección, Empalme, Falsificación, Filtro de Bayer, Filtros de Color, Imagen Digital, Manipulación, Máquina de Soporte Vectorial, Patrón Binario Local, Técnicas Pasivas, Transforma Discreta del Coseno, Transformada Wavelet.

Abstract

The number of digital cameras integrated into mobile devices as well as their use in everyday life is continuously growing. Every day a large number of digital images, whether generated by this type of device or not, circulate on the Internet or are used as evidence in legal proceedings. Consequently, the forensic analysis of digital images becomes important in many real-life situations.

Forensic analysis of digital images is divided into two main branches: authenticity of digital images and identification of the source of acquisition of an image. The first attempts to discern whether an image has undergone any processing subsequent to its creation, i.e. that it has not been manipulated. The second aims to identify the device that generated the digital image. Verification of the authenticity of digital images can be carried out using both active and passive forensic analysis techniques. The active techniques are based on the fact that the digital images have "marks"present since their creation so that any type of alteration made after their generation will modify them, and therefore will allow detection if there has been any possible post-processing or manipulation. On the other hand, passive techniques perform the analysis of authenticity by extracting characteristics from the image. Any editing operation on a given image will leave "traces"that will be used to check the authenticity of the image. Passive techniques are the most widely used since not all devices have the ability to leave "marks.ºn the image to validate its authenticity.

Five passive techniques are proposed in this thesis:

The first combines the analysis of local texture patterns with the use of the intrinsic characteristics of the compression and format of each of the blocks of an image to extract a group of singularities with which to form a vector. This will later be used in a vector support machine to create a model that will make it possible to verify the authenticity of the image, identifying different types of manipulations such as Copy-Move or Splice.

The second estimates, without previous training, the interpolation pattern used by the sensor of the camera that generated the image, making it possible to locate the inconsistencies present in the image that arises in a subsequent editing process such as Copy-Move or Splice type manipulations.

The third, based on JPEG compression, analyzes the error level that highlights the pixels that have a different compression level from the rest of the image and marks them as not original image content, detecting splice-type manipulations.

The fourth, which is based on chromatic interpolation algorithms, divides the image into overlapping blocks and estimates the interpolation pattern and the mean square error, also detecting splicing type manipulations.

The fifth and final one uses the discrete cosine transform. Through these coefficients the features, that form the transfer vectors used, are obtained. These vectors are grouped together and by using a tolerance threshold, duplicate areas can be identified in an image.

All these techniques have been evaluated using different image datasets, public and private, demonstrating their validity as they represent an advance in the state of the art passive techniques of forensic image analysis.

Finally, it should be noted that these techniques have been implemented in a forensic analysis tool, developed within a project of the European Union's Framework Programme for Research and Innovation Horizon 2020, a tool that has also been validated by police in various European countries in a real environment.

Keywords: Authenticity, Bayer Filter, Chrominance, Colour Filters, Copy-Move, Counterfeiting, Detection, Digital Imaging, Discrete Cosine Transformation, Discrete Wavelet Transform, Features, Forensics, Local Binary Pattern, Manipulation, Passive Techniques, Splicing, Vector Support Machine.

Part I

Research Description

Chapter 1

Introduction

The widespread use of digital cameras on mobile devices is a reality in everyday life. Images generated by mobile devices can be seen daily in news, e-mail or social networks. Websites such as Facebook, YouTube or Twitter, among others, are at the top of the list of most visited websites, with a considerable part of their content captured with digital cameras on mobile devices. This means that in certain cases there are legal restrictions or limitations on their use in places such as schools, universities, government offices, businesses, etc. In addition, and as a consequence of all the above, everyday digital images are more often used as silent witnesses in legal proceedings (child pornography, industrial espionage, street violence, social networks, etc.), is in many cases crucial pieces of evidence of a crime.

On the other hand, digital images are modified using specialized editing software that increasingly automates and facilitates the work of those who use them. Tasks such as copying and moving a region of a photograph to hide something or someone from the scene can be done with very few mouse clicks and with finishes that are virtually imperceptible to the human eye. This makes forensic analysis of digital images increasingly important.

1.1. Problem Identification

As described above, the reliance on digital images as a potential source of information poses a significant problem. The main factors contributing to this mistrust are:

- The existence of professional tools for post-image processing at low cost which are available to both novice and expert users and allow for the creation of realistic and compelling counterfeit images.
- The increasingly common practice of image exchange by Internet users.

To address this problem, forensic techniques have been developed to verify the authenticity of digital images, detecting any manipulation of their content. These techniques can be classified into two categories [Con11][FCD12][ZMM+19]: active and passive.

Active tampering detection techniques are the most intrusive as they are based on the use of signatures and watermarks that are stored in the image at the time of its creation. Specialized hardware and software are therefore needed to create and store this information.

In contrast, passive techniques for detecting manipulations have no prior knowledge of the image or its characteristics. These techniques use the information of the image content to detect the traces left by the manipulations made on it. Passive methods, therefore, have a wider range of application and are more useful for detecting and locating manipulations in images.

Passive techniques are in turn classified by the type of analysis they perform: those based on analysis of image metadata, those based on analysis of pixels to find patterns that leave traces of manipulations, and those based on analysis of physical factors such as light to identify inconsistencies in the image.

1.2. Motivation

Although many passive techniques have been proposed in literature to validate the authenticity of digital images, the emergence of increasingly robust and user-friendly editing tools creates a persistent need for more effective forensic image manipulation detection techniques that address the challenges facing forensic analysts today.

The main challenges that motivate this thesis are the detection of multiple clone regions, the location of clone areas, the robustness of the forensic technique against geometric transformations and how noise and compression from the manipulation has an effect on the resulting image.

The research of this thesis starts with the study of the existing techniques in literature and focuses on improving some of the existing limitations which currently exist.

1.3. Objectives

The general objective of this thesis is focused on the creation, adaptation and improvement of forensic techniques that speed up the verification of the authenticity of digital images. More specifically, the research focuses on the detection of cloning-type manipulations of regions of an image and of the splice-type manipulations of digital images. These two types of manipulations are chosen because they are the most widely used today. To achieve this goal, the following specific objectives are proposed:

1. To investigate the strengths and weaknesses of existing forensic manipulation detection techniques in literature.

2. To develop a technique to classify manipulated images from multiple media outlets (Internet, social networks, USB, external drives).

3. To analyze and develop passive techniques to locate splice-type manipulations in digital images.

4. To analyze and develop passive techniques that detect cloning and locate manipulated regions in digital images.

1.4. Summary of Contributions

The results of the research carried out in this thesis include various contributions that have been published in international journals indexed in the Journal Citation Reports and in specialized congresses in the area. As represented in Figure 9.1, these contributions are framed in the forensic analysis of digital images. More specifically, they are focused on so-called passive techniques for detecting digital image manipulations.

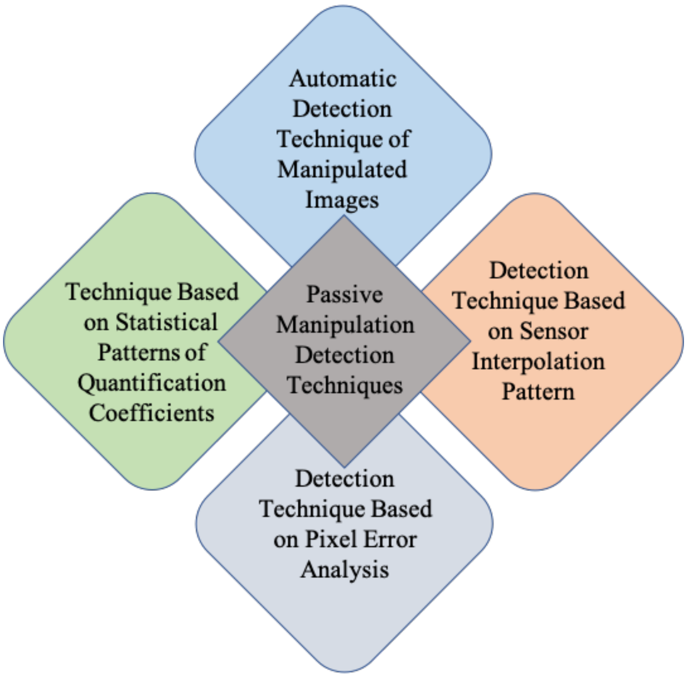

Automatic Detection Technique of Manipulated Images
Technique Based on Statistical Patterns of Quantification Coefficients
Passive Manipulation Detection Techniques
Detection Technique Based on Sensor Interpolation Pattern
Detection Technique Based on Pixel Error Analysis

Figure 1.1: Thesis contributions

The first contribution of the thesis, presented in Chapter 4, is a technique of automatic classification of manipulated images. The proposed technique combines the analysis of local texture patterns and image compression characteristics with the use of supervised learning techniques to verify the authenticity of an image, identifying Copy and Move and Splice type manipulations [AVSOVHC18] [AVQHSOGV19]. Once the images are identified as manipulated, the following four contributions of this thesis analyze them individually to locate the region subjected to the alteration, specifying four passive techniques based on the analysis of the pixels at different levels of depth to find patterns that leave traces of the manipulations.

The second and third contributions constitute Chapter 5, which describes two techniques that detect splice-type manipulations by means of a heat map. The first one analyses the error level that highlights the pixels that have a different degree of compression from the rest of the original image, identifying them as non-original image content. The second determines whether or not there is a modification in an image by estimating the interpolation pattern and the mean square error [AVMHPA+19] [AVGFSOGV20b].

Thus, the fourth contribution, described in Chapter 6, consists of a technique for detecting manipulations in digital images that estimates the interpolation pattern used by the sensor of the camera that generated the image. This is in order to identify the inconsistencies present in the image left by the editing process after performing Copy and Move and Splice type manipulations [AVGFSOGV20a].

Finally, the fifth contribution, which makes up 7, proposes a technique that detects manipulations of the Copy and Move type within an image from the statistical patterns of the quantification coefficients of the Discrete Cosine Transform [AVMHPA+20].

All the techniques proposed in this thesis have been evaluated using, in addition to the most used image datasets in the literature, their own dataset with a variety of scenarios.

Finally, all these techniques have been integrated into a forensic analysis tool developed within a project of the European Union's Framework Programme for Research and Innovation Horizon 2020, a tool that has also been validated by police in several European countries in a real-life environment.

1.5. Outline of the Thesis

This thesis is structured as follows:

Chapter 2 presents some basic concepts about digital imaging, its creation process and the components involved. It then describes the types of manipulations in digital images, what their origins were, and what forensic techniques exist, detailing each of them to gain a better understanding of this work.

Chapter 3 discusses the most current forensic techniques for identifying manipulations in digital images, offering an in-depth explanation of the components involved in the forensic analysis of digital images.

Chapter 4 describes the first contribution of this work, namely a technique for identifying manipulations in images based on characteristics extracted from the wavelet transform, the discrete cosine transform and the local binary pattern of the images [AVSOVHC18] [AVQHSOGV19].

Chapter 5 specifies the technique for identifying manipulations in Splice type digital images, which estimates the interpolation patterns and consequently allows the identification of the anomalies of this pattern present in the spliced sections of the manipulated image [AVGFSOGV20a].

Chapter 6 develops two techniques for identifying and locating Copy Move and Splice manipulations. The first is based on the mean square error of the image interpolation pattern and the second on the analysis of the error level in the JPEG compression of the image [AVMHPA+19] [AVGFSOGV20b].

Chapter 7 shows a technique for detecting Copy and Move manipulations. The technique uses the characteristics extracted from the overlapping blocks of the image using the discrete cosine transform [AVMHPA+20].

Finally, Chapter 8 contains the main conclusions of this work and the future lines of work arising from it.

1.6. Reader of this Thesis

This thesis does not require the reader to already possess advanced knowledge of the content discussed. Several definitions and concepts available in the literature are repeated in order to make it independent. However, the reader should bear in mind that it is important to have basic theoretical and practical knowledge about forensic analysis of digital images, statistical calculations, wavelet transformation and computational algorithms.

Some of the problems involved are discussed here, while for others, it is considered that the reader already has a foundation for understanding the concepts discussed. The bibliography provides the reader with additional information to further explore the study carried out in this thesis.

Chapter 2

Image Forgery Techniques

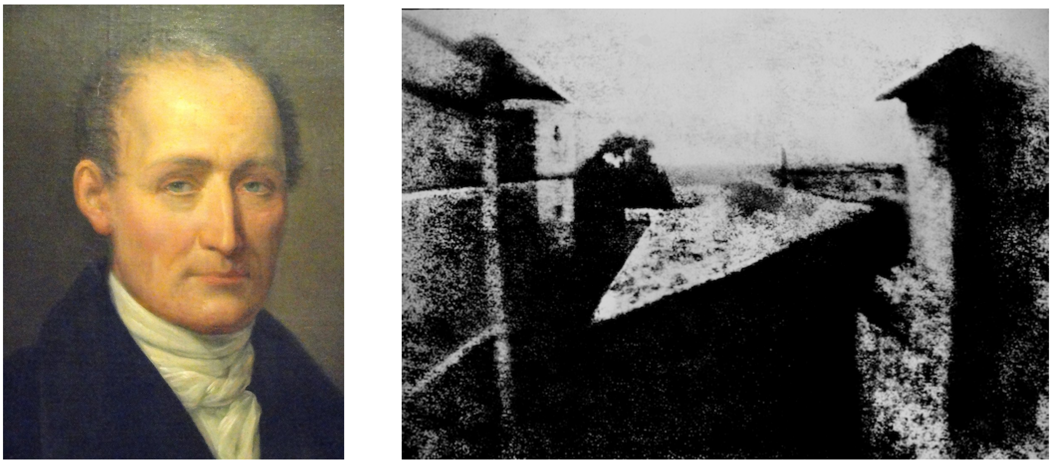
Nowadays, we use the Internet in almost all of our everyday routine. Each of those common activities has been replaced with a mobile app or web app. Activities such as grocery shopping, buying tickets to go to the movies, talking and interacting with family and friends or getting an update on the local and international news can now be easily done by using one or two applications from our computers and mobile devices. Due to the growing problem of the "fake news" phenomenon, a big concern arises how to identify and control the fake or manipulated information that circulates on social media networks

Section 2.1 of this chapter covers the history of the development of photography up to the present day, as well as a description of the process of creating a digital image and each component involved in it. Then, Section 2.2, displays examples of image manipulation throughout history, images that have travelled the world and at one time caused controversy. Section 2.3 introduces the main techniques of image manipulation. Each of these techniques is explained in this chapter to serve as a basis for the reader to better understand the contributions made in this thesis. Finally, Section 2.4 summarizes all the chapter.

2.1. Digital Images

The first photograph in history, of which there are records, is the "Point de vue du Gras" and was taken by Joseph Nicéphore Niépce, a French engineer in 1826, Figure 2.1. The picture was recorded after several hours of exposition and "printed" over a pewter sheet (zinc alloy, lead and tin) [Mar90]. Since then, the image "creation" process has changed, making it easier to capture and print the scene.

One of the milestones in the history of photography was the development of the first 35mm film Single-Lens Reflex (SLR) camera. After the invention of the 35mm film in the late 1800s by William Kennedy Dickson and Thomas Edison [DD00], several inventors tried to use the film for photography instead of motion pictures.

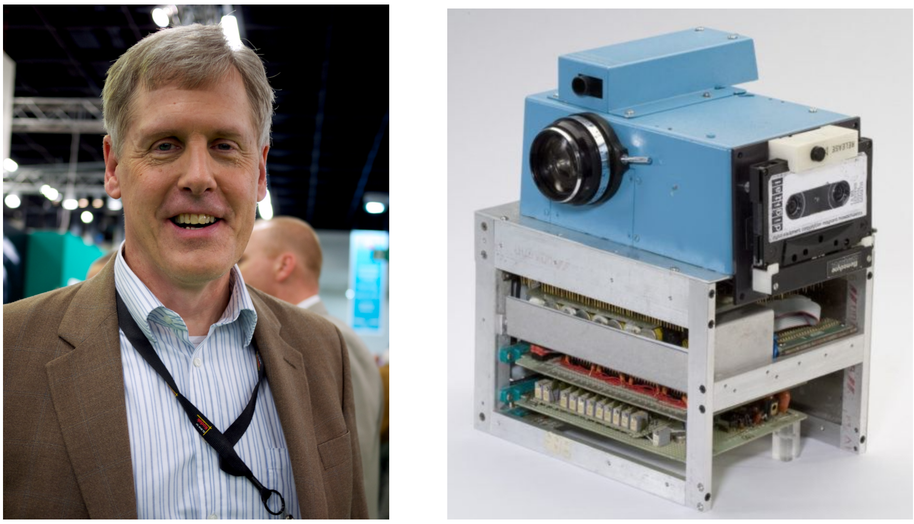


(a) *Joseph-Nicéphore Niépce* (b) *Point de vue du Gras*

Figure 2.1: Inventor of the photography process and the world's first photography

The first patent issued for a 35mm camera was to Pau Audouard, Albert Lleo and Claudi Baradat in 1908, but without any commercial production or sale. However, it wasn't until 1913 that the single-lens reflex camera became accessible and widely available. The first 35mm reflex camera was the American Tourist Multiple [SVOZMR18].

In the digital era, Steven Sasson invented the world's first digital camera in 1975, Figure 2.2. The invention was made during the time that Sasson worked for Kodak, but they didn't see the potential of digital photography at that time. Digital cameras, also known as a Digital Single-Lens Reflex (DSLR), have evolved very fast and have almost entirely displaced regular SLR cameras from the market.



(a) *Steven Sasson* (b) *World's first digital camera*

Figure 2.2: The first digital camera and its inventor

To understand image forgeries and its artifacts, the following sections, of this chapter, will briefly cover the concepts and the components involved within digital image processing pipeline, before jump into image forgery techniques.

2.1.1. Image Processing Pipeline

Regarding the image processing pipeline, digital photography is a modification of classical chemical photography. In a traditional 35mm film camera, light enters through the lens, and this causes a chemical reaction on the film, as it is composed of light-sensitive chemicals. On the other hand, modern digital cameras use the same principle. However, the light does not act on any film, but on an electronic element called a sensor, in this case, the phenomenon is no longer chemical but electronic.

In digital photography, a group of components, within the source camera, equipped with electronic photodetectors are used instead of a photosensitive film. In this case, developing the film to obtain and print the captured image is not part of the process, but rather the picture is generated and saved onto a storage device (external or internal), and can easily be seen on the camera's screen or on a computer (Figure 2.3).

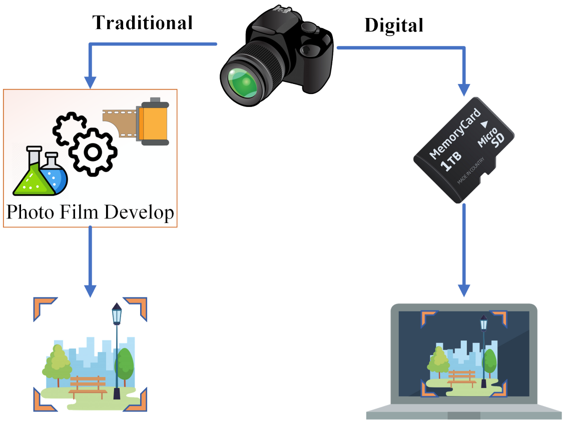


Figure 2.3: Traditional and Digital photo viewing process

During the digital image processing pipeline, components such as the lens, filter and, sensor carry out the work to generate an image and save it within the memory of the camera, Figure 2.4 shows the components and its order of "participation" during the process.

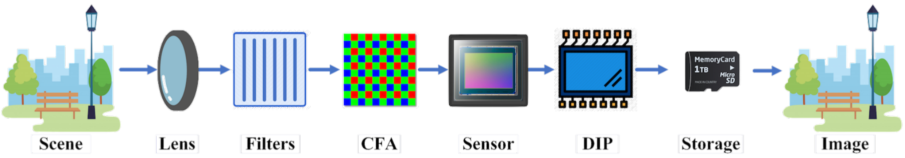
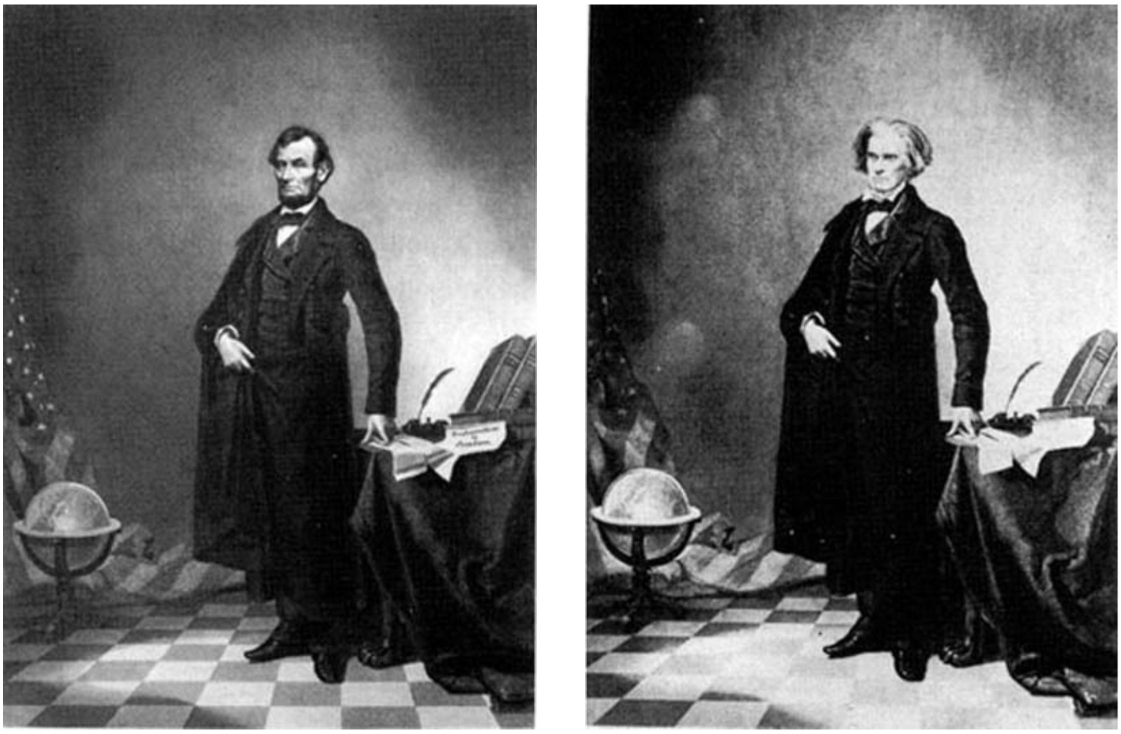


Figure 2.4: Image pipeline process' components

2.2. Image Forgery Across Time

Image forgery began almost at the same time that Joseph Nicephore created the first photograph in 1826.A first example of manipulation was recorded in the early 1860s, when a photo of Abraham Lincoln was composed by inserting Lincoln's head onto the body of a portrait of John C. Calhoun (Figure 2.5).



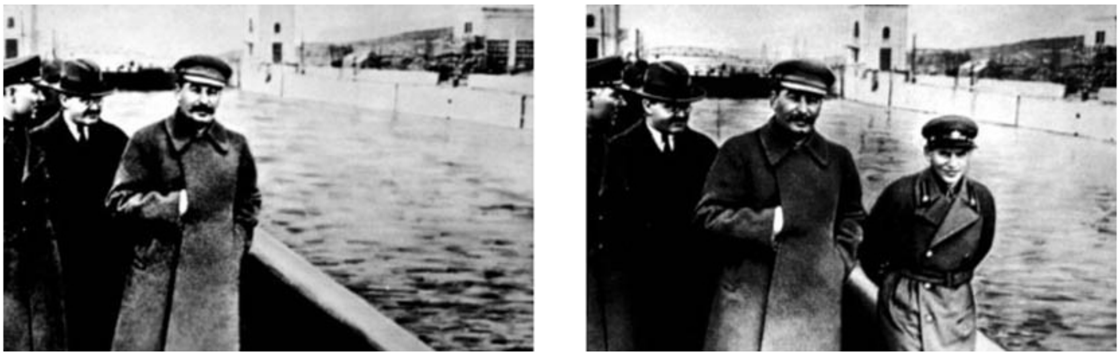
(a) *Forged Image* (b) *Original Image*

Figure 2.5: Abraham Lincoln's manipulated photo

In 1920 Joseph Stalin made use of photographic retouching for propaganda purposes. As can be seen in the Figure 2.6, the Commissioner for Internal Affairs, Nikolai Yezhov, was made to disappear after his execution in 1940. Another manipulated historical photograph took place in 1937, when Adolf Hitler had Joseph Goebbels removed from the original photograph (Figure 2.7). It is not yet clear why Goebbels was removed from the image.

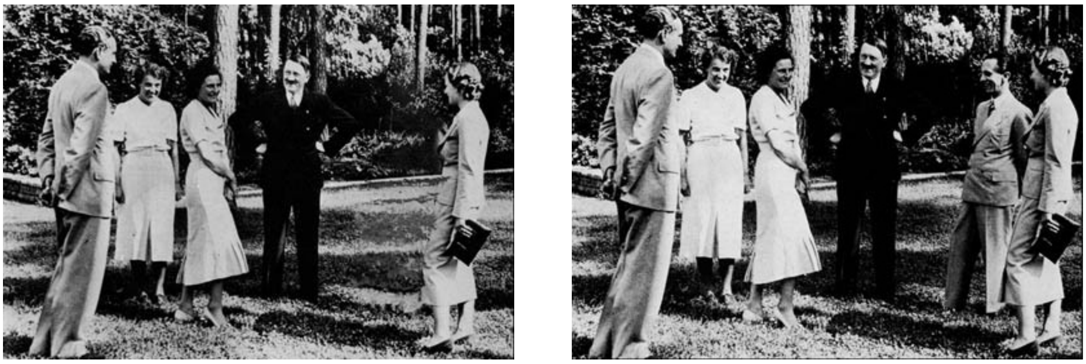
Benito Mussolini had an assistant removed from the original photograph in 1942, thus achieving a more heroic portrait of himself and in turn a better propaganda message (Figure 2.8).

In the 1980s, the digital era began to emerge and the first digital image editing tools appeared. Among the first known manipulations of this period is that of National Geographic on the cover of their magazine on Egypt, which took place in 1982 (see Figure 2.9). Gordon Gahen took a horizontal image of the Great Pyramids of Giza, which had to be scaled to fit the magazine's vertical format. The photographer who took the picture complained to the magazine about the manipulation of his image.



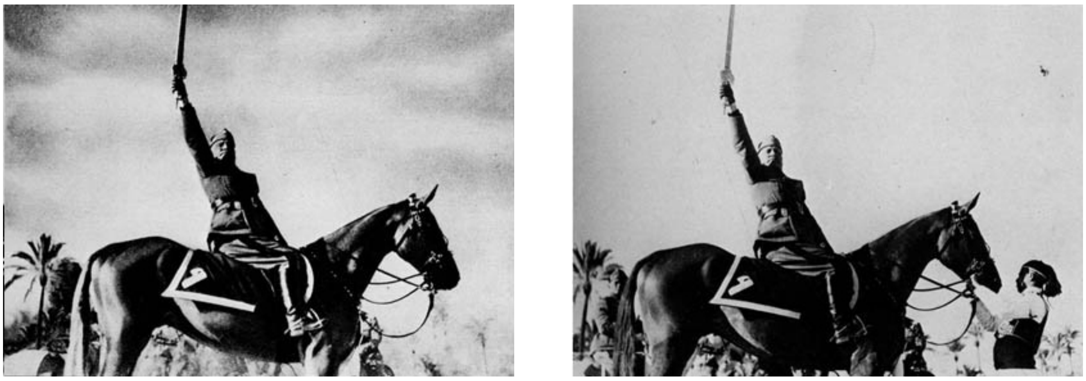
(a) *Forged Image* (b) *Original Image*

Figure 2.6: Joseph Stalin's manipulated photo



(a) *Forged Image* (b) *Original Image*

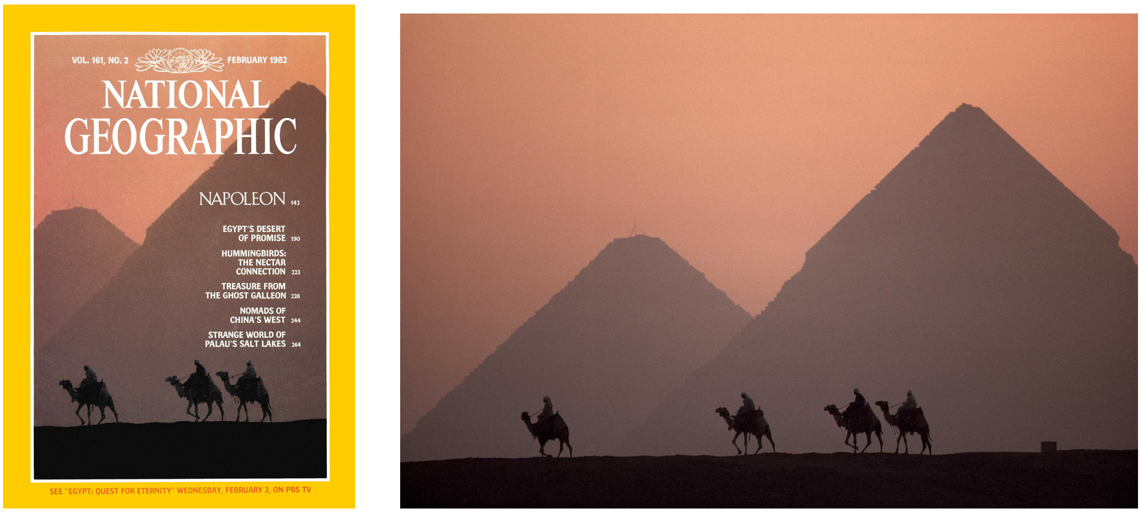
Figure 2.7: Disappearance of Joseph Goebbels from the original image



(a) *Forged Image* (b) *Original Image*

Figure 2.8: Benito Mussolini's propagandistic image

Another famous example of journal-related manipulation dates back to 1989. The cover of TV Guide shows this image of daytime talk show host Oprah Winfrey. This image was created by splicing Winfrey's head onto the body of actress Ann-Margret (Figure 2.10(a)), taken from a 1979 advertising shot (Figure 2.10(b)). The compound was created without

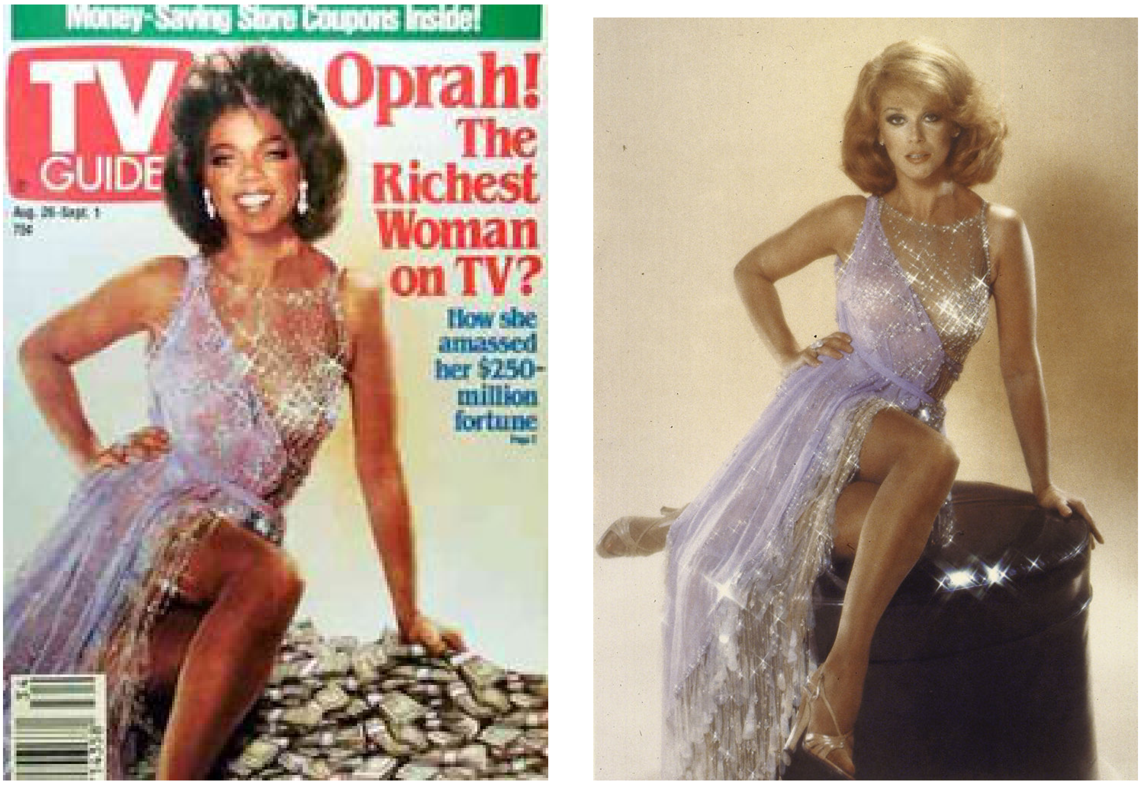

VOL. 161, NO. 2
FEBRUARY 1982
NATIONAL GEOGRAPHIC
NAPOLEON 142
EGYPT'S DESERT OF PROMISE 190
HUMMINGBIRDS: THE NECTAR CONNECTION 223
TREASURE FROM THE GHOST GALLEON 228
NOMADS OF CHINA'S WEST 244
STRANGE WORLD OF PALAU'S SALT LAKES 264
SEE "EGYPT: QUEST FOR ETERNITY" WEDNESDAY, FEBRUARY 3, ON PBS TV

(a) *Forged Image*

(b) *Original Image*

Figure 2.9: Cover of National Geographic magazine manipulated

the permission of Winfrey or Ann-Margret.

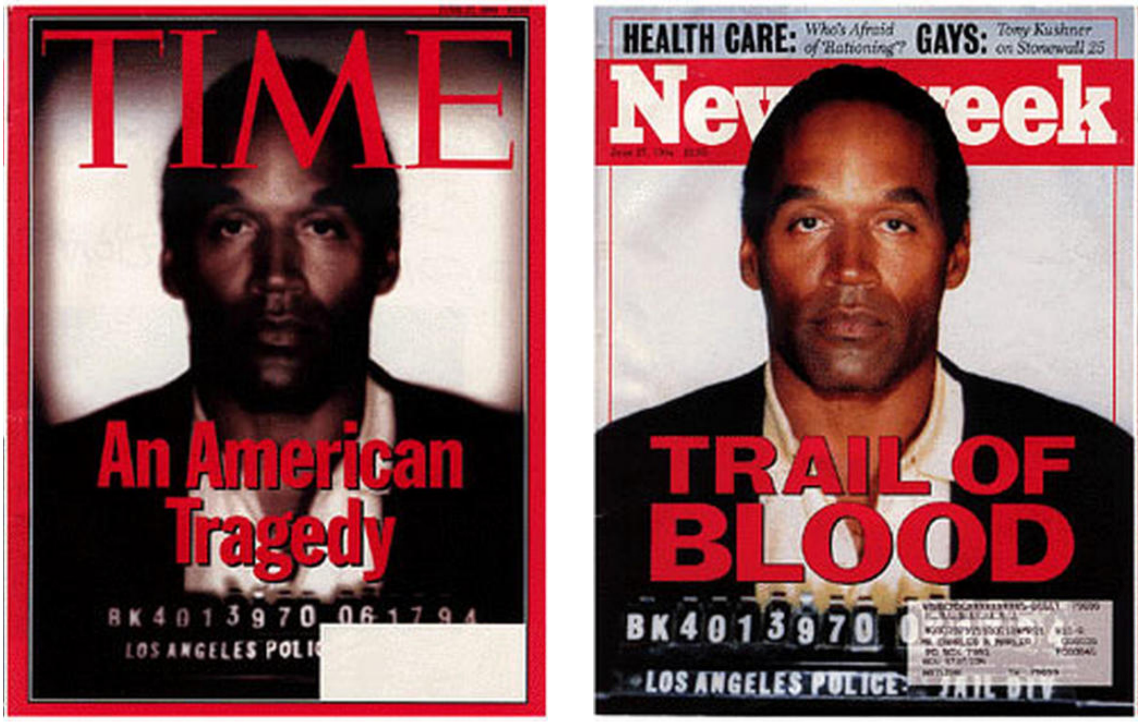

Money-Saving Store Coupons Inside!
TV GUIDE
Aug. 26-Sept. 1
Oprah! The Richest Woman on TV?
How she amassed her $250-million fortune

(a) *Forged Image*

(b) *Original Image*

Figure 2.10: Manipulated TV Guide's cover

In the 1990s, high-resolution digital cameras, powerful personal computers, and sophisticated photo-editing software arrived, and photographs were increasingly manipulated. In turn, detecting tampering became more costly. The manipulation carried out by Time magazine in 1994 stands out as an example of this decade. On the magazine's cover is an image of Simpson shortly after he was arrested for murder. This image was digitally manipulated from the original photograph that appeared, unaltered, on the cover of Newsweek (Figure 2.11(b)). The magazine was later accused of manipulating the photograph to make Simpson appear more threatening and dangerous, as shown in Figure 2.11(a).



(a) *Forged Image* (b) *Original Image*

Figure 2.11: Manipulated image of O.J Simpson on the cover of Time magazine

Nowadays, it is difficult to highlight manipulated images due to the large volume of these circulating on the network without any kind of filter. Since the advent of social networks it has become possible for anyone to quickly share a photo with a large number of users. It is very common to find a predominance of manipulated images on these networks as they allow users to cause a greater impact on their followers. In addition, some of these networks provide integrated image manipulation tools that allow for editing right after the photo is captured.

2.3. Image Forgery Techniques

Nowadays, thanks to technological advances, modifying an image is relatively trivial. Using specialised image editing software is becoming increasingly simple, and its results are challenging to detect with the naked eye. But what is a manipulated image? A manipulated image is the result of altering the content, format or any intrinsic feature of the initially conceived image.

There are different types of image manipulations, and each one is grouped according to the kind of feature it alters within the image, as the Figure 2.12 shows.

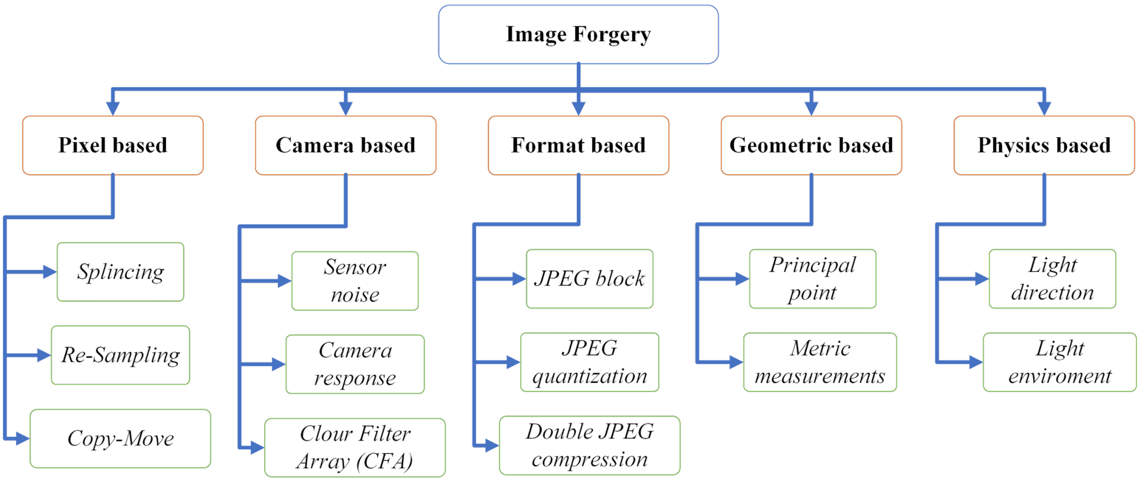


Figure 2.12: Main image forgery techniques [NKK15][CR19]

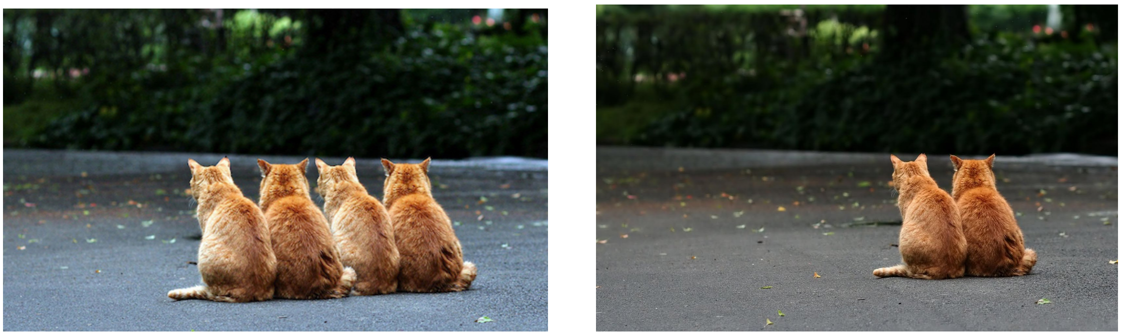
2.3.1. Copy-Move

Copy-Move manipulation is typically performed with the aim of making an object “disappear” from the original image by covering it with a small fragment copied from another part of the same image. This method is also used to duplicate existing objects in the image. As these copied blocks come from the same image all their characteristics will be compatible with the rest of the content making it very difficult for the human eye to detect them.

When the copied region is moved, it is usually accompanied by the “blurring” effect generally used on the edges of the modified region in order to reduce the irregularities between the original and the manipulated region.

Detection techniques for this type of manipulation focus on the search for duplicated areas, although if combined with other post-processing techniques, such as the application of colour filters, it can be quite difficult to detect using existing techniques.

An example of this technique is shown in Figure 2.13. In the manipulated Figure 2.13(a) the two animals shown in Figure 2.13(b). have been duplicated.



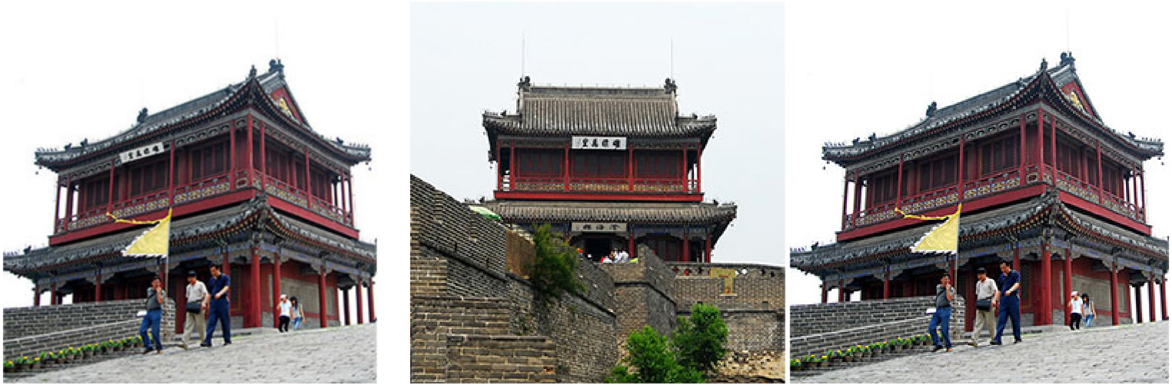
(a) *Forged Image* (b) *Original Image*

Figure 2.13: Example of an image manipulated with Copy-Move technique

2.3.2. Image Splicing

The splicing technique consists of copying a region of a certain image and pasting it onto a different one, so that both images are mixed, consequently creating one image. It is widely used in photomontages where two images are combined giving the sensation of being one. Detecting the exact area that has been forged in the image by means of the splicing technique is of great complexity in comparison to previous manipulation techniques. This is because it is not possible to look for duplicated areas since the manipulated region comes from a different image.

An example is shown in Figure 2.14 where two original images can be seen that have been used to create a spliced photo. From the first image shown in Figure 2.14(b), the temple sign has been cut out and pasted into the second image in Figure 2.14(b) to generate the final result that can be seen in Figure 2.14(a).



(a) *Forged Image* (b) *Original Images*

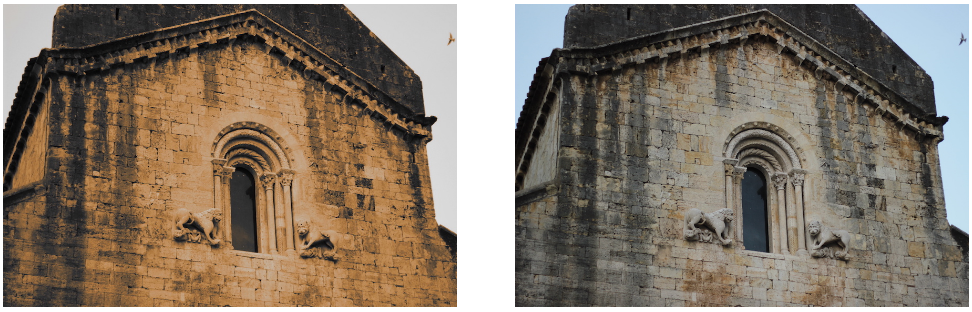
Figure 2.14: Splicing Image

2.3.3. Image Retouching

This manipulation is one of the most used for its simplicity. Almost all digital image editing software incorporates a selection of pre-defined filters to be automatically applied to the image (see Figure 2.15). Its aim is to improve the final finish of the image by modifying aspects such as tone, saturation, brightness, contrast, etc.

They do not have to involve a "malicious" change in the content of an image but it is mentioned in this work because it is important to take it into account during the detection of any other manipulation, as it is likely to affect the operation of the method used.

Aesthetic retouching consists of applying small modifications to the original image without copying any area of the rest of the image or storing it from a different one. It aims to perfect finishes or hide imperfections for aesthetic purposes while maintaining characteristics similar to those of the original image. The most commonly used tools in this type of manipulation are sanitizing, profiling, smearing, blurring and highlighting (see Figure 2.16).



(a) *Forged Image* (b) *Original Image*

Figure 2.15: Example of a manipulated image by a filter technique

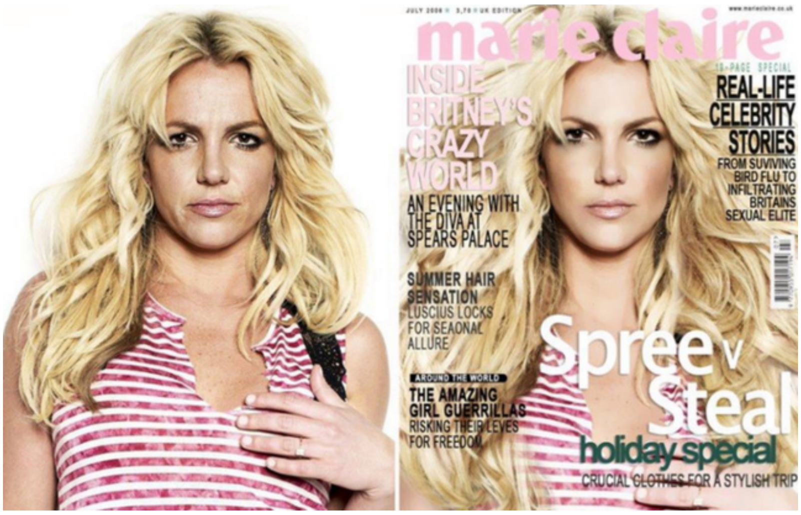

marie claire
INSIDE BRITNEY'S CRAZY WORLD
AN EVENING WITH THE DIVA AT SPEARS PALACE
SUMMER HAIR SENSATION
LUSCIUS LOCKS FOR SEASONAL ALLURE
AROUND THE WORLD
THE AMAZING GIRL GUERRILLAS RISKING THEIR LIVES FOR FREEDOM
REAL-LIFE CELEBRITY STORIES
FROM SURVIVING BIRD FLU TO INFILTRATING BRITAINS SEXUAL ELITE
Spree v Steal
holiday special
CRUCIAL CLOTHES FOR A STYLISH TRIP

Figure 2.16: Example of aesthetic retouching

2.3.4. Image Fingerprint Manipulation

During the process of generating an image it is possible that some defects are introduced and are reflected as noise in the final image. The defects are produced by the device's sensors and are very helpful in identifying the camera that generated a particular image.

There are various sources of imperfections and noise introduced in the different stages of the image generation process in the camera. Even if a uniform and fully lit photograph is taken, it is possible to observe small changes in intensity between the pixels. This is due to shooting noise that is random and largely caused by the noise pattern that is deterministic and remains almost the same if several pictures are taken of the same scene.

There are two types of noise that are important. The first type of noise is caused by defects in the camera's sensor. These include point defects, hot spot defects, dead pixels, unexpected effects, and so on. These defects cause the pixel values in the image to deviate greatly. For example, dead pixels appear as black spots in the image and hot spot defects appear as very bright pixels. The noise pattern in an image refers to any spatial pattern that does not change from one image to another. Dark streams are streams lost from the sensor substrate to individual pixels. This varies from pixel to pixel, and the variation is known as fixed pattern noise Fixed Pattern Noise (FPN). Photo Response Non Uniformity (PRNU) consists mainly of pixel uniformity Pixel Non-Uniformity (PNU) and low frequency defects such as zoom settings and light refraction on dust particles and lenses. This pattern is very important to detect the source of an image as each device will have a different noise pattern associated with it [KMC^{+}07] [GVSORCHC17]. In the Figure 2.17 we can see a general scheme of the types of noise pattern of the sensor.

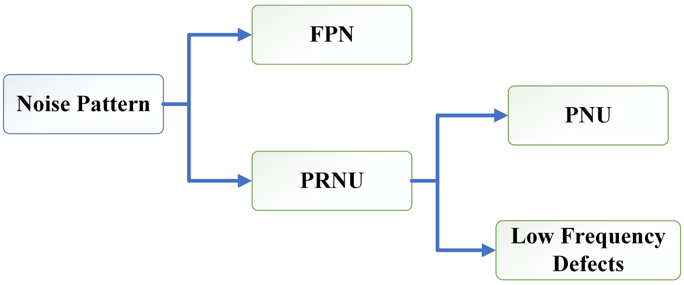


Figure 2.17: Types of sensor noise pattern

The manipulation techniques that make use of the fingerprint differ from the previous ones in that they do not focus on the graphic part of the image but on the information it contains. These techniques can be divided into two types that pursue different objectives [GVSORCHC17], these types are:

- **Image Anonymization**: It consists of removing the information from the original image source. This information also known as the image source is the relationship between a device and the image it generates. In this way it is possible to identify the individual owner of the device that generated the image, so it is very important to know this information. To perform this type of technique, the PRNU is removed from the image to be modified, so that the resulting image is of anonymous origin.

- **Image Falsification**: It is based on making a modification to the origin of the image. In this way it is possible to change the real information on the model and brand of the device that generated the digital image. One very common use of this technique is to extract the fingerprint of an image (PRNU) and replace the fingerprint of another image with the extracted one. Consequently, it is possible to falsify the information about the origin by changing the device that actually generated the image for the print for another one that did not. Most anti-forensic source identification techniques make use of this pattern.

2.4. Summary

The use of sophisticated editors has changed the length of time and prior knowledge required to execute any of the techniques explained in this chapter. Therefore, identifying a “fake image” becomes harder, especially, since the current software editors create a very realistic result and hide any artefact related to the possible manipulation. This chapter has covered the main image forgery techniques to help understand this thesis’ contributions regarding the comprehension and interpretation of an image forgery and the different types of such that exist and how they work. Considering the most used and popular forgery techniques, this thesis has focused on Copy-Move and Splicing forgeries.

Chapter 3

Image Forgery Detection Techniques

In relation to the previous chapter, after having explained each of the main image manipulation techniques, it is necessary to detail and explain the main techniques for detecting these manipulations. This chapter will cover each of the techniques to detect all the manipulation techniques shown in the previous chapter. To this end, Section 3.1 explains the taxonomy of different approaches to forensic image analysis. Then, the state-of-the-art techniques related to the contributions of this work compose each of the following sections. Section 3.2 presents the most representative works of techniques to identify Copy-Move manipulation. In Section 3.3 the techniques that identify image splicing are covered and in Section 3.4 identification techniques of visual retouching in images are presented. Section 3.5 presents state-of-the-art methods in identifying double compression in images. Finally, Section 3.6 summarises everything discussed in this chapter.

3.1. Image Forgery Detection Taxonomy

Within multimedia forensics, image forgery analysis can be classified into two main approaches [CFGL08][MS10][SDJ+18]: active and passive. These two approaches use different methods and techniques to achieve their goals.

Active forensic analysis techniques have been proposed mainly to verify the ownership and authenticity of audio-visual material and thus detect copyright violations [CMB+07][SA17][NL19][FLJ+20]. The techniques used in the active approaches are mainly based on the analysis of digital watermarks, and signatures left by the device during the image generation. In general, watermarks are imperceptible traces that are added to the multimedia file during its creation. By verifying their presence in the file, it is possible to identify their origin and authenticity. The main requirements that active forensic image analysis techniques must meet are robustness, imperceptibility and capacity.

Nonetheless, one of the major drawbacks of this type of approach is that many cameras do not have the ability to incorporate such watermarks or signatures, so their scope is limited and therefore this restricts the application of such methods.

On the other hand, passive approaches analyse the content and characteristics of the image without any prior information. The techniques based on this approach focus on the analysis of any inconsistencies in the image features such as noise, Camera Response Function (CRF), Color Filter Array (CFA), etc.

The approaches and their techniques used in digital image forgery detection are shown in Figure 3.1. Due to the techniques proposed in this thesis are part of the passive approach, such techniques principally used in the passive analysis with be analysed below.

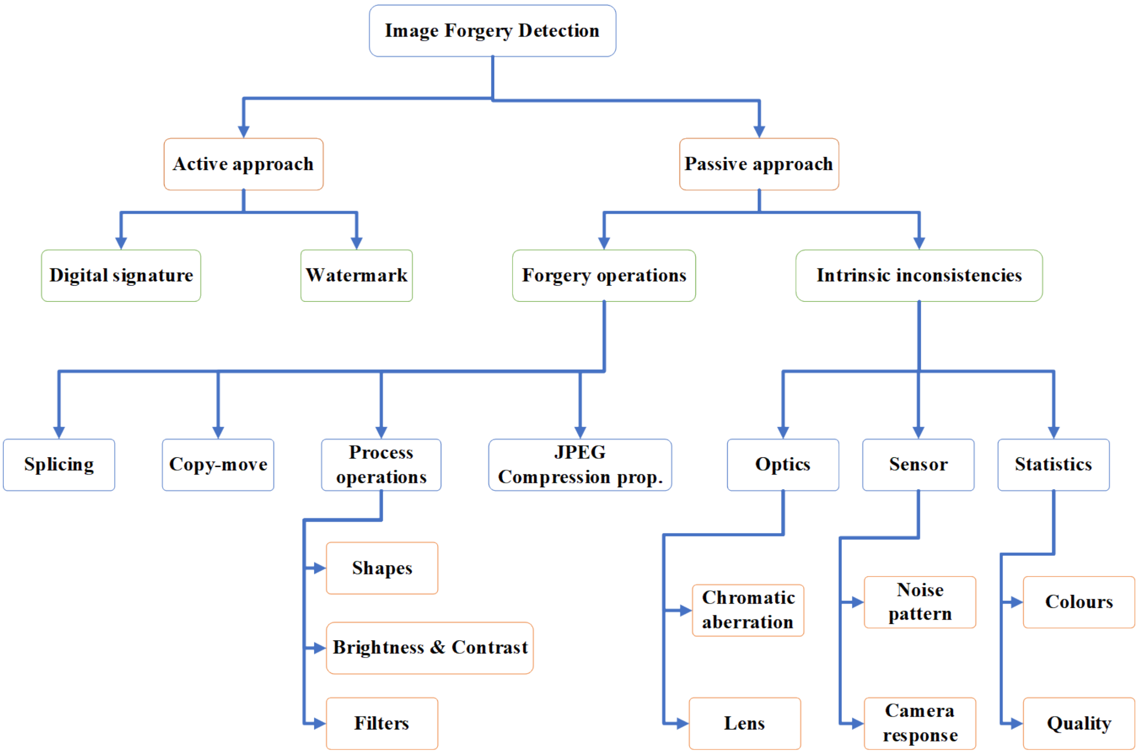


Figure 3.1: Image forgery detection approaches and techniques

3.2. Copy-Move Detection Techniques

The copy-move technique is another popular method used today for image forgery in which a region of an image is used to hide another region from the same image. The existence of two identical regions is not ordinary occurrence in natural images; thus, this property can be used to detect this type of manipulations. Even after applying some post-processing processes, such as edge smoothing, blurring, and adding noise to eliminate visible traces of manipulation, there will be two extremely similar regions in the manipulated image.

In the literature a large number of copy-move forgery detection methods have been proposed. Nevertheless, all of these methods can be classified into two main categories: block-based and keypoint-based methods [PC18][TU19]. Of all of these methods, one of the most used to detect copy-move forgery is using a block matching algorithm. In this algorithm, the image is divided into overlapping blocks, and the blocks are compared to find the duplicated region. Figure 3.2 shows a general scheme of the block matching algorithm.

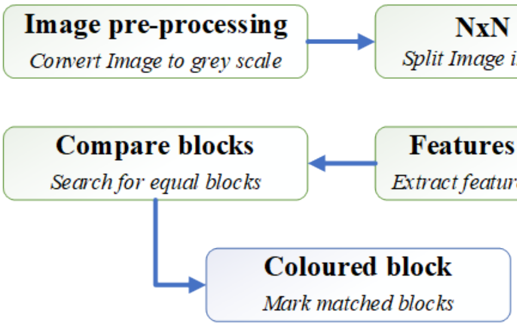


Figure 3.2: Diagram of copy-move forgery detection techniques

Fridrich *et al.*, in [FSL03] proposed a method based on the DCT to identify copy-move forgery. The method split the image into overlapping blocks of 16 × 16. Then, the DCT coefficient characteristics are extracted from each block and these coefficients are classified lexicographically. After the lexicographical classification, comparable squares are distinguished, and the duplicated regions are found. Fridrich *et al.* present one of the first method that use DCT to detect copy-move forgeries on images.

In [CPF04], Popescu *et al.* proposed an algorithm to identify duplicate regions in a digital image. Their algorithm applies Principal Component Analysis (PCA) instead of DCT. PCA is applied on small fixed-size image blocks and each block is then lexicographically sorted. This proposed technique has shown high efficiency in detecting copy-move forgeries and also, shows that the detection is possible even in the presence of significant amounts of corrupting noise.

In [KW08], the use of Singular Value Decomposition (SVD) was proposed to distinguish the altered areas in a digital image in 2008. With SVD, the feature vector is also extracted, and the dimensions thereof reduced. Similar blocks were identified through the use of lexicographic classification. This method proved to be robust and efficient. The experimental results demonstrate the validity of the proposed approach for manipulated images subjected to Gaussian blur filters, noise contamination and compressions.

In [HGZ08] proposed to detect copy-move forgery in digital images using the Scale-invariant Feature Transform (SIFT) algorithm in 2009. The authors presented the SIFT calculation algorithm using the block matching function. This algorithm offers excellent results even when the image is compressed or noisy.

In [BJGY10] a scheme based on Speeded Up Robust Features (SURF) was proposed, which have better key point characteristics than SIFT because they work more effectively with post-processing techniques such as brightness and blur variations. However, the methods based on key points present a problem of visual output because the copied and moved regions consist of lines and points that do not show a clear and intuitive visual effect.

In [ABC+11], Amerini *et al.*, introduced a method based on SIFT features. This method can detect duplicated regions in images. Also, the method proposed can detect which geometric transformation was applied. Given that the copied region of the image looks the same as the original, the key points extracted in the duplicated region will be identical to those in the original. This method is also useful with low-quality factor compressed images.

Muhammad *et al.* in [MHB12] presented a blind copy-move forgery detection method based on the Dyadic Wavelet Transform (DyWT). The algorithm mainly uses two kinds of information, the similarity between blocks of the image and the noise inconsistency between these parts. The experiments were executed in three different scenarios: i)same sized images, and copy-move region without rotation, ii)different sized images and copy-move region with and without rotation, iii)different quality (Q) factors. The results have shown that the algorithm works better than some previous proposals.

In [ZG13], Zhao *et al.*, proposed a method based on DCT and SVD which include seven steps to analyze and detect duplicate regions in images. Firstly, the input image is divided into overlapping blocks, then DCT is applied to each block, and the DCT coefficients are quantized. Later, each quantized block is divided into non-overlapping sub-blocks, and SVD is applied to each sub-block. Then, features are extracted to reduce the dimension of each block using its largest singular value. Finally, all feature vectors are lexicographically sorted, and duplicated image blocks are matched by the predefined threshold. The experiment has shown that the proposed algorithm not only detects copy-move forgery and locates the duplicated regions, but can also analyse and detect manipulation of images with Gaussian blurring, additive white Gaussian noise, and JPEG compression.

In [PC18], Park *et al.* proposes an approach that can manage several geometric transformations, including rotation, scaling and reflection. The proposed algorithm use keypoints and descriptors from the image based on the SIFT to analyze possible reliable matched pairs by using the distance ratio between the most and second most similar match. The matched pairs are then included in a set of real-matches sorted by their ratio value.

Table 3.1 presents a summary of the copy-move detection techniques analysed by comparing their results in terms of accuracy.

Table 3.1: Comparison of copy-move forgery detection techniques

Work	Used method	Observations
[FSL03]	DCT coefficients and lexicographic classification	Robust to image retouching
[CPF04]	PCA, *Eigen* values and lexicographical classification applied	Good results against compressed or noisy images
[KW08]	SVD and lexicographic classification	Validity against blur, noise and compression filters
[HGZ08]	SIFT calculation algorithm using the block matching function	Good results against compressed or noisy images
[BJGY10]	Based on SURF feature descriptors	Works well with post-processing techniques such as brightness and blur variations
[ABC$^+$11]	Extraction of key points with SIFT algorithm	Effective in compressed images with a low quality factor
[ZG13]	DCT coefficients and singular value decomposition (SVD) features	Effective over Gaussian blurring, additive white Gaussian noise, and JPEG compression.
[PC18]	Extraction of key points with SIFT algorithm	Effective over several post-processing transformations such as rotation, scaling, JPEG compression, and additive white Gaussian noise.

3.3. Splicing Detection Techniques

Image splicing is a common and relatively easy task to perform, and many modern tools provide ways to conceal the modification by applying further post-processing operations, leaving no visible traces. Splice detection can be addressed in many ways, for example, by detecting signal differences in the original background and the spliced fragment, or by detecting post-processing operations applied to borders.

When splice manipulation is performed, the local distribution of the edge micro-patterns is altered by introducing new micro-patterns into the moved region. Therefore, it changes its regularity and the local frequency distribution. All of the methods discussed below differ only in the way they model the structural changes caused by counterfeiting. The characteristics with which the different images are trained and classified, that is the representation of the changes made to an image, is the determining factor in the level of success of a method of image manipulation.

Most algorithms for splicing detection are split into sub-processes, which are common to each other, as shown in Figure 3.3.

In [SCC07], Shi *et al.*, proposed a blind, passive image splice detector method. This method, extracts statistical features from the images, and also, the resulting features of applying Multi-size Block Discrete Cosine Transform (MBDCT). These two groups of features build the feature vector that will be the input for the Support Vector Machine (SVM) classifier. The experiments carried out by the authors show a higher detection rate, up to 90% accuracy. The public dataset used during their experiments was "*Columbia*[NC04]".

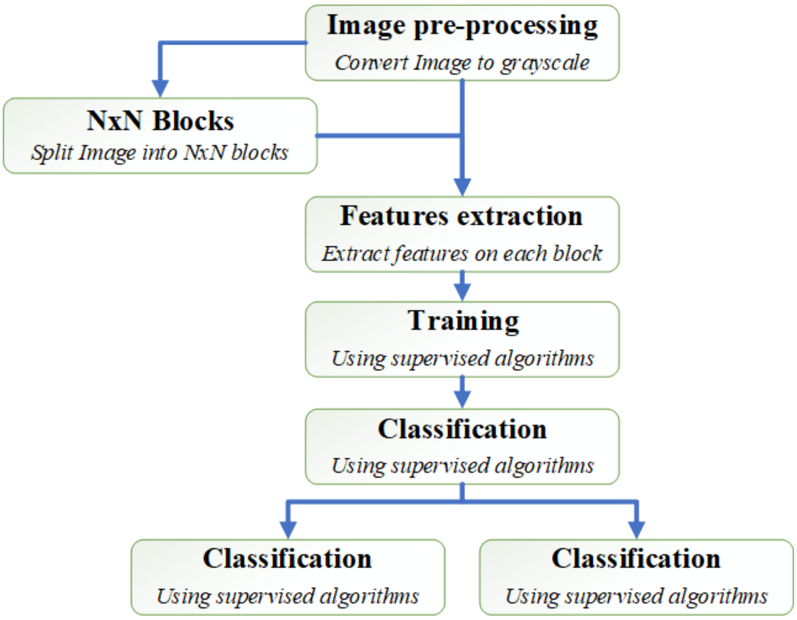


Image pre-processing
Convert Image to grayscale
NxN Blocks
Split Image into NxN blocks
Features extraction
Extract features on each block
Training
Using supervised algorithms
Classification
Using supervised algorithms
Classification
Using supervised algorithms
Classification
Using supervised algorithms

Figure 3.3: Process diagram of splicing detection techniques

In [ZKR08] Zhang *et al.*, presents an algorithm to classify spliced images. The author's algorithm uses the characteristics extracted from 2D matrices generated when applying MBDCT [SCC07]. Their work, beside previous research, introduce as features, the image quality metrics Image Quality Metrics (IQM). The new vector built from all those features is the input for the SVM classifier. The dataset used for the experiments was "*Columbia*". The obtained accuracy was up to 87.10%

Wang *et al.*, in [WDT10] proposed a passive image tampering detection method based on modelling edge information. Because the human eye is more sensitive to the luminance component (Y) than the chroma component, some tampering artifacts left in the chroma channel are undetectable at first sight. Therefore, the Wang *et al.*'s algorithm transforms the image from Red-Green-Blue (RGB) to Luminance-Chroma-Components (YCbCr) space colour and uses only the Cb and Cr components to extract the edge information. Moreover, a finite-state Markov Chain (MC) is used to model the defined edge image and to capture its interpixel dependencies. Once the features are extracted, a nine-dimensional vector is built to be the input of the SVM classifier. The experiments carried out have shown that the proposed algorithm is very useful for tampering detection. The accuracy obtained was up to 95.6% with the public dataset "*CASIA TIDE v2.0* [DWT13]".

In [ZLLW11], Zhao *et al.* compared the effect of using different space colors on the detection of image splicing. The authors made a comparison of the YCbCr space color against the regularly used RGB. The algorithm extracted and used four gray level run-length run-number Run-length Run-number (RLRN) vectors from the chroma channels. After the feature extraction, the resulting vector is the input to an SVM, which is the algorithm classifier. The experiments used the datasets "*CASIA TIDE v1.0*" and "*Columbia*" and the detection effectiveness was up to 94.7% of accuracy. The results show that the chrominance channels are more effective than RGB in detecting forgery within images.

Xia *et al.*, in [XYS+16], introduced an algorithm to identify forgery within fingerprints images. To extract the needed features to build the input vector for the classifier, Xia *et al.*'s algorithm uses the DWT and LBP. The accuracy obtained by the experiments was shown to be up to 92%. The images used in [XYS+16] are in the "*LivDet*" [Lis07] dataset.

In [AH17] Alahmadi *et al.*, presented a method based on DCT and LBP to detect splicing and copy-move counterfeits. Their algorithm pre-processes the image by changing the space color to YCbCr. Then, the Cb and Cr components are divided into overlapping blocks and LBP is applied to each block. Each block is transformed into the DCT domain and their DCT coefficients are extracted to build the feature vector. As a classifier, the authors used an SVM. The experiment results show an accuracy of detection up to 97.77%. The used dataset was "*CASIA TIDE v2.0*".

3.4. Retouching Operations

Image processing is the manipulation of images using digital computers [dSM05]. Its use has been growing in the last decades and its purposes vary from medicine to entertainment. The action domain of this type of image manipulation is to rotate, scale, filter or adjust brightness and contrast of an image.

In [KF11] Kee *et al.*, proposed an algorithm to detect the use of post processing image retouches to enhance the images in magazine covers. To do that, the authors used a dataset composed of 468 images and introduced a metric (range 1–5) for quantifying alterations of the image done by digital photo-editing techniques, depending on the amount of image alteration. The photometric and geometrical modifications of the original and the retouched photo were calculated for each picture. Then, eight statistics were extracted – four statistics from the mean and standard deviation of the motion magnitude calculated individually on the face and body and four statistics from the means and standard deviations of both the spatial boundaries of local smoothing/sharpening filters and the Structural Similarity Index Metric (SSIM) – incorporating the degree of photographic retouch to calculate the correlation with the evaluation of each photo. In the experiments a SVM was used to calculate the degree of modification of the image. The absolute prediction error was below 0.5 and 1.0 for 81.4% and 99.1% of the images, respectively.

The detection of image sharpening is one of the main topics in image forensics, and the most popular sharpening method is the Unsharp Mask (USM) [Mal77]. Ding *et al.* in [DZY+15] proposed a technique, to detect image USM sharpening, which used the overshoot artifacts left by the USM algorithm on the image's edge pixels. The author's method extracts features from the edge perpendicular binary coding histogram to train an SVM, and then, defines whether an image was sharper using USM or not. Despite that, the experimental results show that Ding *et al.*'s method outperforms existing methods, although it still remains at a low level for weak image sharpening. To improve the results from their previous work, Ding *et al.* in [DZDS18] the authors proposed a new method that is capable of detecting weak USM sharpening. The new method differs from the previous one by introducing further steps such as the edge direction, the edge areas definition using interpolation algorithms and the local threshold calculation before building the histogram and extracts the features to feed the SVM to be trained. The experimental results show a 97.85% of accuracy, which is a superior performance compared with similar algorithms.

Virtual makeup is a notable problem in facial images. This specific type of image processing simulates real makeup on one face within the image, which can delay the process of identification, for example, of a crime suspect. It's because of this that several methods to automatically detect virtual makeup on images have been proposed.

In [CDR13] Chen *et al.*, proposed an algorithm to detect automatic facial makeup in images. The proposed solution extracts features from the shape, texture, and colour of three predefined facial regions. Chen *et al.*'s algorithm has five stages which are: i)face detection and landmark localization, ii)face normalization, iii) Region Of Interest (ROI) extraction, iv)feature extraction and v) feature classification. The experiments were conducted in two face datasets (YouTube Makeup (YMU) and Makeup in the Wild (MIW) [DCR12]), and using an SVM with kernel Radial Basis Function (RBF) showed an accuracy of up to 93.5% in makeup detection and up to 95.45% of overall classification rates.

Kose *et al.* in [KAD15], proposed a facial makeup detector algorithm to reduce the impact of makeup on face recognition further. This study work with Chen *et al.*'s datasets (YMU and MIW [DCR12]). The main differences between Kose *et al.*'s algorithm and Chen *et al.*'s are the use of grayscale images as input and the analysis of the entire face without extract information from facial subregions. The author's experiments were carried out using the YMU dataset as a training dataset of the SVM and the MIW as a test group. The achieved accuracy was up to 98.5% which is significantly higher than the 93.5% obtained by Chen *et al.*

When a retouched image is compared with the original one, the face identification is considerably degraded. Due to this, in [BSVB16] Bharati *et al.* proposed a supervised algorithm that used Boltzmann machine to detect retouching in face images. Moreover, to evaluate the proposed approach the authors introduced two face image databases with unaltered and retouched images (ND-IIITD). The authors compare their supervised method with similar state-of-the-art algorithm proposed by Kee *et al.* in [KF11] and the obtained results show a better performance by Bharati *et al.*'s approach. While Kee *et al.* proposal gets a correct classification accuracy of slightly less than 50%, the Boltzmann machine supervised algorithm proposed by Bharati *et al.*, gets an 87.1% which is a great improvement in the accuracy detection.

3.5. Image Double Compression Detection Techniques

JPEG is one of the most popular and commonly used compression formats for digital images [BM13]. In the early years of digital photography, the majority of digital cameras exported their pictures in a JPEG format, and nowadays, almost all devices with a built-in digital camera generate and save their images using this format. Moreover, when a forgery technique such as splicing or copy-move is applied over an image, the double JPEG compression is inevitable. Then, identifying whether a JPEG image has been re-compressed or not is an important matter when a forensic analysis is conducted.

In [HHS10] Huang *et al.*, proposed a method to detect JPEG double compression on images with the same quantization matrix. Their approach is based on the observation of how the JPEG coefficients monotonically decrease between the first and second JPEG compression of the image. The authors use a random perturbation strategy to discriminate the difference between a single and a double compressed image, especially when the image is compressed with a relatively high-quality factor. Their experiments show a direct relationship between the accuracy and the quality factor of the test image.

The intrinsic property of decreasing the number of JPEG coefficients, left by two consecutively compressed images with the same quantization matrix, can be used to detect double JPEG compression [HHS10]. Nevertheless, if the JPEG images have been compressed with low-quality factors, then, detecting double compression in this scenario becomes challenging to achieve. Niu *et al.* in [NLZN19] proposed an approach that enhances Huang *et al.*'s method and makes it capable to detect double compression on images compressed with low-quality factors. The main difference between [HHS10] and Niu *et al.*'s method lies in the random selection of +/- 1 valued JPEG coefficient of the recompressed image. The experiments show that the method increased the accuracy of up to 1.74% compared with previous algorithms, especially when the quality factor is less than 90.

3.6. Summary

In this chapter the main techniques for identifying image manipulations have been presented. The techniques presented in this chapter are state-of-the-art in forensic image analysis. The advance of technology allows today to alter images using specialized editing software that increasingly automates and facilitates the work of those who use them. Tasks such as Copying and Moving a region of a photograph to hide something or someone from the scene can be done with very few mouse clicks and with finishes that are practically imperceptible to the human eye.

Just as technology has facilitated the way in which modifications can be made to an image, so has it facilitated the access to a wider audience for these images. The benefit of unrestricted access to information, and much more, the benefit of unrestricted sharing, can be jeopardized if the information being shared is not the true one. In these times, when it is necessary to be connected and, above all, informed; identifying possible biases or direct, intentional manipulations of the information we receive is fundamental.his is why it is necessary to constantly improve and develop techniques that allow us to identify the increasingly sophisticated methods of altering images used to present them as real.

Chapter 4

Automatic Image Forgery Detection Technique

This chapter proposes a digital image authentication technique that combines the analysis of local texture patterns with the discrete wavelet transform and the discrete cosine transform to extract features from each of the blocks of an image. Subsequently, it uses a vector support machine to create a model that allows verification of the authenticity of the image. The experiments were performed with falsified images from public databases widely used in the literature that demonstrate the efficiency of the proposed method.The chapter is structured into four sections. Section 4.1 briefly describes the methods used in the technique. Section 4.2 specifies in detail the proposed technique. The experiments conducted to evaluate the functionalities and robustness of the technique are described in Section 4.3. Finally, Section 4.4 is a summary of what is presented in the chapter.

4.1. Overview

To understand the proposed technique, it is important to know the concepts on which the technique is based. In this section, these concepts will be explained.

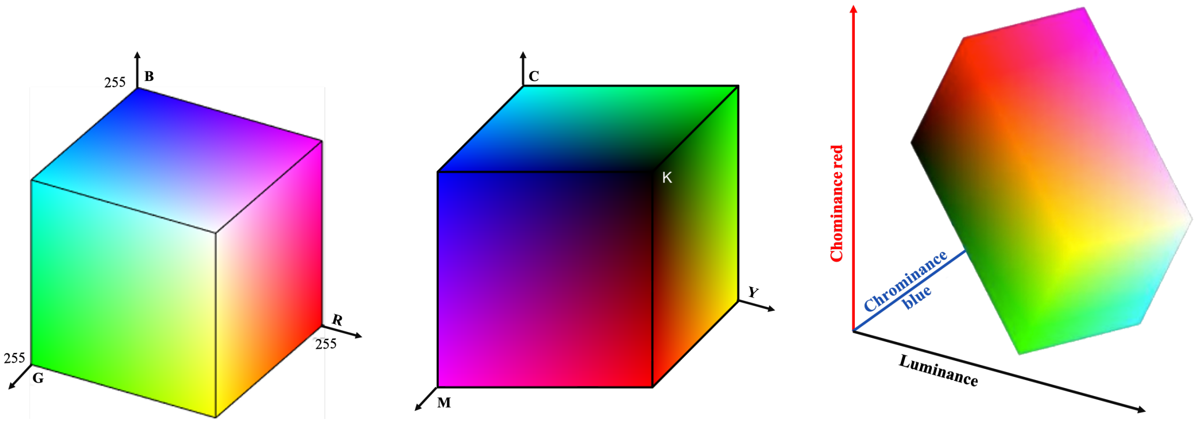
4.1.1. Colour Models

The colours of an image can be represented in multiple ways. Each of these representations is called a Colour model. In [WC04] the colour model is defined as a system to measure colours that can be perceived by humans. A colour model assigns numerical values to different colour components of an image. Colour models are generally classified into three or four components. Among the most common are the RGB, Cyan-Magenta-Yellow-Key (CMYK) and YCbCr models.

- **RGB Model**: It is one of the best-known models due to its widespread use in computers, screens, televisions and mobile devices, among others. It is based on the representation of a colour by mixing by adding the colours red, green and blue. This model assigns values between 0 and 255 to each of its three components (red, green and blue) and the sum of the three will form the value that represents the colour. A digital image is formed by a multitude of pixels, each one of them will have its own value within the model RGB being 0 the colour black and 255 the colour white.

- **CMYK Model**: This model is based on representing a colour in the components cyan, magenta, yellow and black. This representation differs from the previous model is done in a subtractive way and not by addition. This is why a fourth component is necessary to represent the correct black tone. The values assigned to each component range from 0 to 100, depending on the desired intensity of the tone of the component. This model is widely used in the printing sector as it proportions a good contrast in the different tones used to print. Therefore, before the printing of an image, the model RGB is usually transformed into the model CMYK.
- **YCbCr Model**: It is a colour model used for the processing of digital images, this model is a non-linear coding of space RGB. Colours are represented in the form of luminance and chrominance components. The luminance is represented by the symbol Y and the two chrominance signals are represented by Cb and Cr. Cb places the colour on a scale between the colour blue and yellow. Instead, Cr places it between the colour red and green. Finally, the Y parameter offers the respective information to the black and white colours.

The Figure 4.1 shows a graphical representation to RGB, CMYK and YCbCr colour models.



(a) RGB model (b) CMYK model (c) YCbCr model

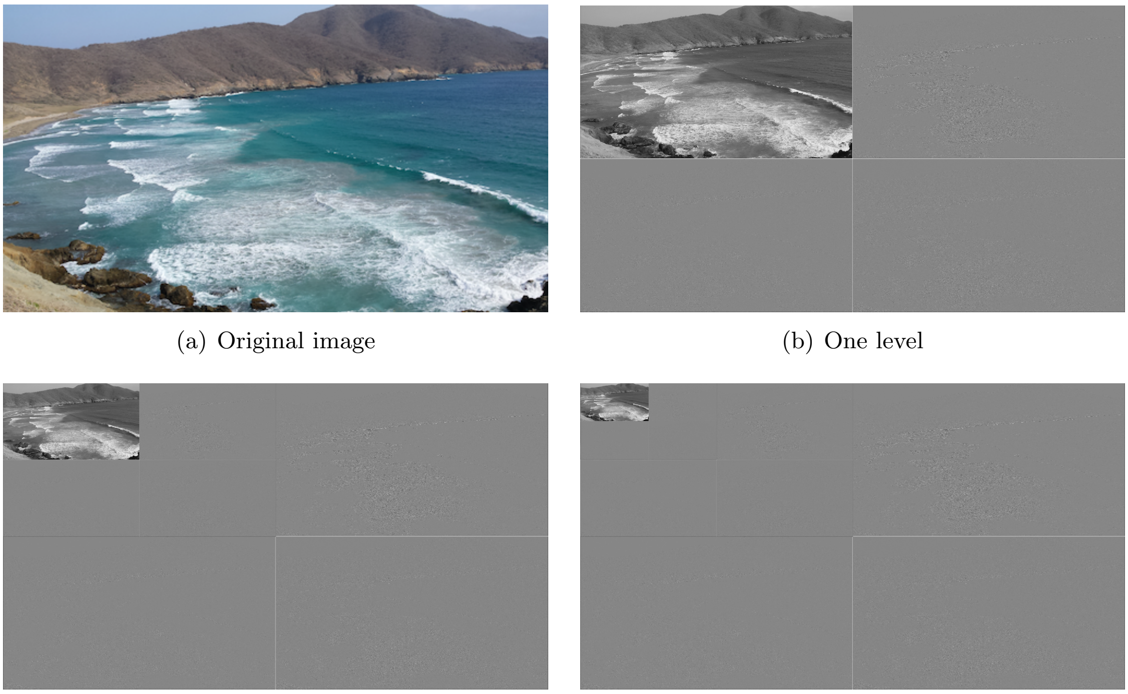
Figure 4.1: Colour models representation

4.1.2. Discrete Wavelet Transform

A wavelet function is a small wave that concentrates its energy over time. It is used to analyse the time and frequency of stationary, non-stationary and time-variable phenomena. The DWT is considered to be the most powerful form of signal representation that can be applied mainly to the processing of signals and images. There are numerous types of wavelet families, the most important are: Haar, Daubechies, Biorthogonal, Coiflets, Symlets, Morlet, Mexican Hat, Meyer, Reverse Biorthogonal, Frequency B-Spline, and Gaussian derivatives [Zha19].

DWT decomposes the signal in the time domain using two filters, a high-pass filter and a low-pass filter. These two filters produce two different versions of the signal, the low-pass version represents a summary (L) and the high-pass version represents the fine details of the signal (H). Likewise, L and H can be decomposed into an additional level to create a new group of low pass and high pass filters, obtaining the LL, LH, HH and HL sub-bands.

In image processing applications, DWT is applied first to the rows and columns of the image. Therefore, we obtain one approximation image (LL) and 3 high-pass channels corresponding to horizontal detail (LH), vertical detail (HL) and diagonal detail (HH). Figure 4.2 shows an image with its four sub-bands for each DWT decomposition.



(a) Original image

(b) One level

(c) Two level

(d) Three level

Figure 4.2: Example of DWT decomposition in an image to one, two and three levels

The DWT can provide unique and discriminating representations that can quantify vital and interesting structures such as edges and details with good resolution for few coefficients. It is also computationally effective due to the small amount of data with which it works. The final wavelet coefficients can be used directly as characteristics and can be extracted directly from the wavelet domain, describing the anomalies in the image data. Basically, the discrete wavelet transform reduces the correlation between wave coefficients and provides energy compaction in some wavelet coefficients.

4.1.3. Local Binary Pattern

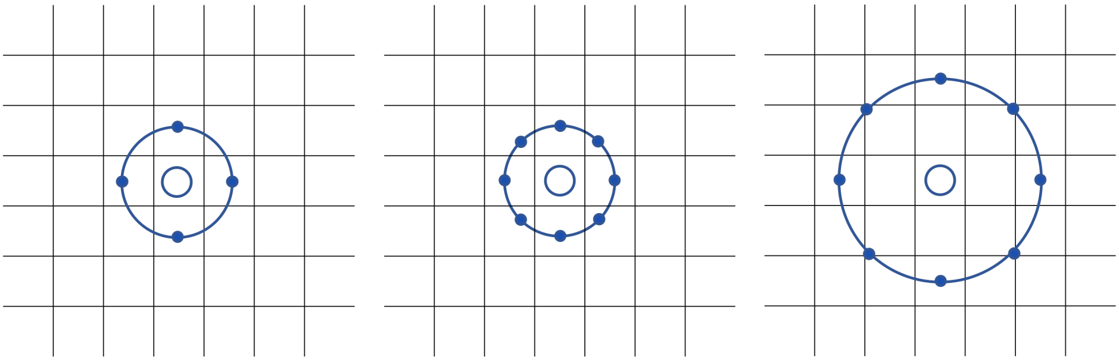
LBP is a local operator that discriminates between different types of textures. When a manipulation is performed, the original texture of the image is distorted. The LBP operator defines a label called LBP code for each pixel of an image [OPM02]. The simplest form of the LBP operator generates a binary code by using 3×3 blocks. Then, LBP works as follow:

To apply LBP in a block $B \times B$ of a gray scale image I_g, first a local neighbourhood LN is defined as a set of the neighbours' pixels P of the image (Equation 4.1).

$$LN = t\ (p_c, p_0, \ldots, p_{P-1}) \tag{4.1}$$

where $p_c = I\ (x, y)$ is a central pixel of the neighbourhood and p_b $(b = 0, \ldots, P-1)$ correspond to the border pixels.

These pixels are Equally distributed in the block forming a circle with radius R $(R \geq 1)$ named circularly symmetric neighbourhood. Figure 4.3 shows different values of P and R for setting the circularly symmetric neighbourhood.



(a) $P = 4, R = 1$ (b) $P = 8, R = 1$ (c) $P = 8, R = 2$

Figure 4.3: Circular symmetric neighbourhood for different (P, R)

To compute the LBP code for $P = 8$ and $R = 1$ ($LBP_{P,R}$), the eight neighbours' pixels p_b are compared with the p_c taking the value s defined according to condition (see Equation 4.2).

$$s = \begin{cases} 1 & , \text{if } p_b \geq p_c \\ 0 & , otherwise \end{cases} \tag{4.2}$$

Once the values of the matrix are obtained, between 0 and 1, the binary number is converted into a decimal number assigning a weight value from weight matrix (w), in clockwise direction, according to Equation 4.3.

$$w = 2^x,\ x \in 0, 1, 2, \ldots 8 \tag{4.3}$$

In this way, the LBP code results from the sum of these values. This value represents the central pixel (p_c). Figure 4.4 shows an example of this process.

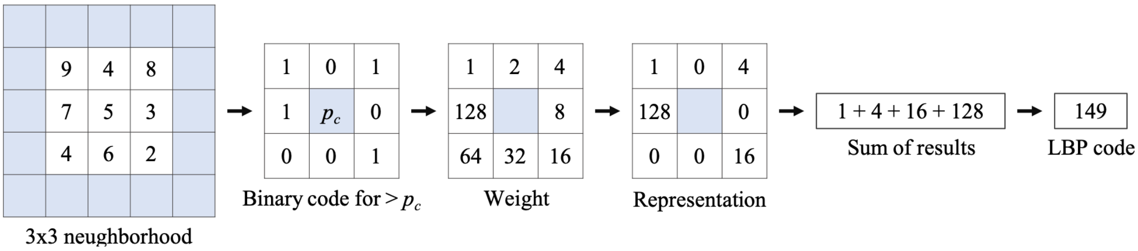


Figure 4.4: Example of LBP code computation for $P = 8$ and $R = 1$

4.1.4. Discrete Cosine Transform

The DCT is a variation of the discrete Fourier transform, where the image is decomposed into sums of cosines, but using only real numbers only. It acts solely on periodic functions with even symmetry and the result is a sequence of real numbers. In the Equation 4.4 the characteristic formula of the DCT-I variant is shown.

$$f_j = \frac{1}{2}(x_0 + (-1)^j x_{n-1}) + \sum_{k=1}^{n-2} x_k \cos\left[\frac{\pi}{n-1} kj\right] \tag{4.4}$$

DCT is used in the detection of manipulations and especially in compression algorithms such as JPEG for its great capacity to compact energy or most of the information in a reduced number of coefficients and for being independent of the number of input data it receives, thus guaranteeing a greater efficiency when working with large images.

DCT also, manages to carry out the compression of images discarding the imperceptible parts for the human eye. This process uses the DCT coefficients to differentiate which points of the image present different characteristics to the rest of which are similar. This is why it is widely used in the detection of manipulations in images. Working with DCT in the chrominance channel allows the obtainment of the coefficients which indicate points of the image that stand out from the rest but which are not visible to the naked eye.

There are eight types of DCT, The most used in the compression of digital images and therefore in the field of detection of falsifications is the DCT-II (Equation 4.5) whose inverse Inverse Discrete Cosine Transform (DCT) corresponds to the type DCT-III shown in the Equation 4.6.

$$f_j = 2\sum_{k=0}^{n-1} x_k \cos\left[\frac{\pi}{n} j \left(k + \frac{1}{2}\right)\right] \tag{4.5}$$

$$f_j = \frac{1}{2}x_0 + \sum_{k=1}^{n-1} x_k \cos\left[\frac{\pi}{n}\left(j + \frac{1}{2}\right) k\right] \tag{4.6}$$

4.2. Technique Description

The main novelty of this approach, over other techniques existing in literature, is its ability to detect two kinds of image forgeries (splicing and copy-move).

The proposed technique consists of three phases: image preprocessing, feature extraction and classification. To have a better picture of these phases, Figure 4.5 shows a diagram of the feature extraction process performed by this algorithm.

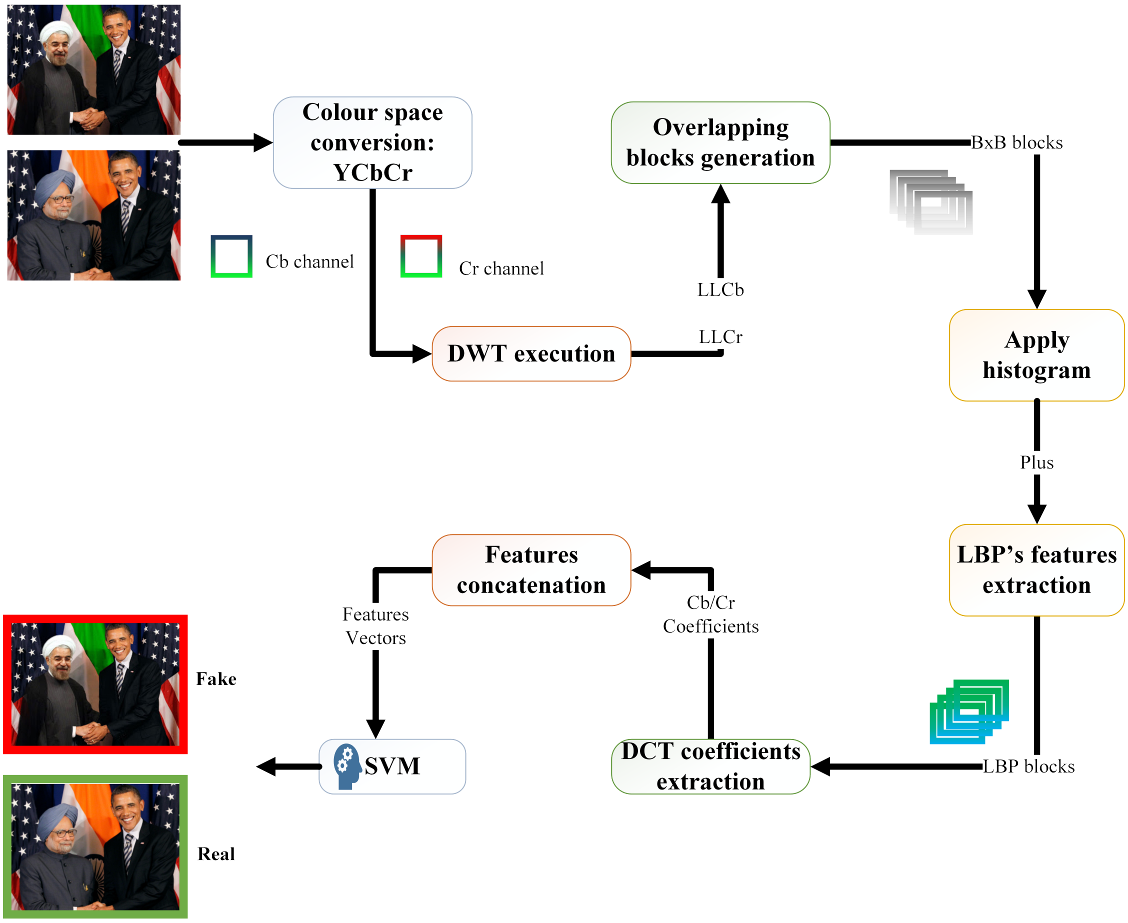


Figure 4.5: Flow diagram of the proposed system

4.2.1. Image Preprocessing

The objective of this phase is to highlight the most important characteristics of the image that best reflect the operations carried out by splice and copy-move counterfeits, and to eliminate unwanted distortions in the Image for further processing. The steps necessary for image preprocessing are described below.

The RGB space is used in the most image forgery techniques. In addition, the attacker applies several filtering operations to hide the forgery traces. The results of some research demonstrate that the YCbCr model is more robust against JPEG compression attacks.

First, as input, each set of N images are received. Each of these $M \times N$ image are converted to the $YCbCr$ colour model using conversion.

$$\begin{aligned} Y &= 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \\ Cr &= 0.713 \cdot (R - Y) + \delta \\ Cb &= 0.564 \cdot (B - Y) + \delta \end{aligned} \tag{4.7}$$

where

$$\delta = \begin{cases} 128 & \text{for 8-bits images} \\ 32768 & \text{for 16-bits images} \\ 0.5 & \text{for floating-point images} \end{cases} \tag{4.8}$$

Then, the component Y is discarded and only the two chrominance components (Cb and Cr) are used.

As human vision notes the luminance component in a clearer way than the chrominance component, it is considered that most of the manipulation traces that cannot be detected by the naked eye can be noted in the chromatic channel. The chromatic channel describes the content of the image signal, such as edges. Thus, any inconsistency in these edges caused by the performance of common forgery techniques (copy-move or image splicing) is emphasized and, therefore, noted.

Once the image is converted, on each Cb and Cr component, the DWT is applied.

Second, each of the components Cb and Cr are then decomposed into four subbands: LL, HL, LH and HH. The discrete wavelet transform analyses an image in different scales and orientations. The splicing process often introduces a sharp transition in the two-dimensional array that represents the image in terms of edges, lines, and corners that are characterised by high-frequency components in the Fourier transform domain. These four subbands are composed by the approximation coefficient LL with the low-frequency information and three address coefficients (HL, LH, HH) with the high-frequency information in different directions (horizontal, vertical and diagonal coefficients, respectively).

Since the LL component concentrates most of the energy, the sub-bands HL, LH and HH are discarded.

Third, it is necessary to apply the Quadrature Mirror Filter (QMF) filter to each of the obtained LL components. The QMF filter is applied in order to get the inverse codification of the frequencies. This filter exchanges the high frequencies with the low frequencies and vice versa [M.09]. Others research about image re-scaling detection has demonstrated successful results using decomposition based on QMF [BM16].

Algorithm 1 shows a summary of steps performed in image preprocessing phase.

Algorithm 1: Image preprocessing phase pseudocode

```
Input: $M \times N$ RGB image
Result: $LL$ component with QMF filter
procedure ImagePreprocessing($I$)
    $I_{(Y,Cb,Cr)} \leftarrow I_{RGB}$;
    Apply DWT decomposition in 4 levels to $Cb$ and $Cr$ components;
    foreach component $c \in \{Cb, Cr\}$ do
        Apply QMF filter to $LL$ component;
    return $LL_{QMF}$
```

4.2.2. Feature Extraction

The feature extraction phase is an important process in image tampering detection. In this section, two methods are combined to identify splicing and copy-move in an image. Local binary pattern and the discrete cosine transform show excellent results in splice detection and copy and move, respectively.

Below, we present the steps that detail this phase:

Once the LL component is obtained and after the QMF filter, the LBP is applied.

First, each component LL_{QMF} is divided into overlapping blocks of $B \times B$ with a sliding window of one pixel and then LBP is applied to each low-frequency component of the chrominance.

From each resulting block, the LBP code is extracted using the following Equation 4.9:

$$LBP_{P,R} = \sum_{i=1}^{P-1} s(p_i - p_c)2^i, \qquad s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{4.9}$$

where P is the number of neighbouring pixels, R is the radius of the neighbourhood and p_c is the value of the central pixel. This operator calculates the LBP code by the value of eight neighbours. Then, the neighbour's binary digits are put together to build a binary code. This process obtains a total of $(M - B + 1) \times (N - B + 1)$ characteristics. The bests results are obtained when the algorithm uses the neighbourhood value of $P = 8$ with radius $R = 1$ for greater accuracy.

The second step is to reduce the number of characteristics obtained with LBP computing the normalised LBP histogram for each sub-band. The resulting histogram can be used as a texture descriptor. In this way, the changes in the different textures are observed with greater efficiency.

The local texture characteristics of the whole image can be described by means of a normalised histogram that is formed by 2^P codes LBP, where P is the number of neighbours that surround the central pixel. For the case of $P = 8$, the total number of features will be 256 codes LBP for each LL_{QMF} component of the sub-band Cb and Cr.

Once the histograms of all the blocks of each component are obtained, the discrete cosine transform (Equation 4.5) is applied to each of them using addhe orthogonal DCT is applied to recover the original signal in the time domain using Equation 4.10.

$$f_j = \sqrt{\frac{1}{n}} x_0 + \sum_{k=0}^{n-1} x_k \sqrt{\frac{2}{n}} \cos\left[\frac{\pi}{n} j \left(k + \frac{1}{2}\right)\right] \tag{4.10}$$

The DCT has a great capacity to compact the energy or most of the information in a reduced number of coefficients and to be independent of the number of input data it receives, guaranteeing greater efficiency when working with images of large dimensions. With this, each block is represented by a set of DCT coefficients.

The result of applying the discrete cosine transform is a finite sequence of points as a result of the sum of several signals with different frequencies and amplitudes. From each set of coefficients, the standard deviation is obtained as a characteristic.

Finally, each initial chrominance (Cb and Cr) component of sub-band LL, from the input image, will be represented as a vector of standard deviations of the sets of coefficients DCT of all LBP blocks. The final vector of characteristics, used in the SVM, will be the concatenation of the vectors of both chrominances as shown in Equation 4.11.

$$Image_{Final\ Feature} = [LL_{QMF}\ Cb_{DCT\ Features}] + [LL_{QMF} Cr_{DCT\ Features}] \tag{4.11}$$

4.2.3. Classification

Machine learning SVM, it is a supervised machine learning technique used for solving pattern recognition and regression analysis problems. SVM allows complex models which are not defined simply by hyper-planes in the input space. To achieve that, the data is mapped into a higher dimension feature space. Once, in the higher dimension feature space, the SVM algorithm divides the data by applying a linear hyper-plane.

Since the machine learning SVM technique has proved to be an accurate and trustful prediction method [SHD14, NYC15, LCWH17]. In the proposed technique of this chapter, a kernelized SVM is used in order to differentiate between authentic and manipulated images.

The SVM will handle a different range of values coming from each step of the algorithm. The LIBSVM framework conducts the pre-processing by scaling all the values in the obtained vector in order to fit all the data in the same range. After that step, from the algorithms' perspective, it doesn't matter if features are continuous or binary, large or small. As a core function, the RBF is used; this function is the most used in similar projects and has shown good results. The optimal values of the parameters from the core of the function RBF (γ, C) are calculated automatically by a cross-correlation process. The library LIBSVM [CL11] has been used, due to its simplicity and efficiency. This implementation of SVM is today one of the most commonly used tools.

4.3. Experimental Results

Throughout this section, all the experiments that have been carried out to evaluate the effectiveness of the proposed training-based manipulation identification proposed technique will be shown.

4.3.1. Datasets

For the evaluation of the proposed scheme, the below public image datasets have been used to carry out the experiments using various image formats and sizes:

- **CASIA v1.0 and CASIA v2.0 [DWT13]**: CASIA v1.0 dataset contains 800 authentic and 925 spliced colour images of size 384 × 256 pixels with JPEG format. All tampered images in the CASIA v1.0 dataset are made only by splicing operation. Spliced images are generated from authentic images by crop-and-paste operation using Adobe Photoshop CS3 version 10.0.1 on Windows XP. CASIA v2.0 dataset contains 7,491 authentic and 5,123 tampered colour images. The images in CASIA v2.0 dataset have different sizes, varying from 240 × 160 to 900 × 600 pixels. Compared to CASIA v1.0, which only contains one image format of JPEG, CASIA v2.0 added uncompressed image samples and also considered JPEG images with different Q factors.

- **DVMMv1 [NC04]**: Its a dataset of 1845 image blocks with a fixed size of 128 × 128 pixels. Develop by the DVMM Laboratory of Columbia University and CalPhotos Digital Library. It has 933 authentic and 912 spliced images.

- **IFS-TC [RPH14]**: IFS-TC dataset belongs to The Image Forensics Challenge competition organized by the IEEE Information Forensics and Security Technical Committee (IFS-TC). The dataset is comprised of several original images captured from different digital cameras with various scenes either indoor or outdoor. The dataset is composed by 875 images divided on 424 "pristine" or "never manipulated" images and 451 "forged" or "fakes" images with Portable Network Graphics (PNG) format. The forgeries included in this dataset are splicing and copy-move.

Table 4.1 shows the summary characteristics of the datasets used in this work experiments.

Table 4.1: Features of the used datasets

Datasets	Type of Tampering	Format	Resolution	Number of Images		
				Original	Fake	Total
CASIA v1.0	Splicing	JPEG	384 × 256	800	921	1721
CASIA v2.0	Splicing	JPEG, BMP, TIFF	240 × 160 to 900 × 600	7491	5123	12614
DVMMv1	Splicing	TIFF	757 × 568 to 1152 × 768	183	180	363
IFS-TC	Copy-move	PNG	1024 × 768 to 3648 × 2736	424	451	875
	Splicing					

4.3.2. Experiments Setup

In all the experiments carried out, *Python* has been used as a programming language, due to its great flexibility to perform data analysis and its high speed in handling input and output.

The forgery detection is a problem of two classes: Authentic images and manipulated images. Therefore, to carry out the training and classification of the data, a support vector machine has been used since it has an excellent performance when working with problems of two classes. Since the proposed scheme uses an SVM as a classifier system, the SVM system was previously trained with 100 genuine images and 100 forged images.

All experiments have been repeated 10 times to remove the randomness problem on the images selection for training and testing process. For each execution, 50% of training and testing samples are randomly changed from each dataset. Finally, the final test result is the average of ten tests performed. Finally, the final test result is the average of ten tests performed.

The characteristics of the equipment in which the experiments were carried out are presented in Table 4.2. It is an important factor to take into account since the execution times of the different tests vary according to the resources available.

Table 4.2: Features of the experimentation equipment

Resources	Features
Operating System	Ubuntu 17.04
Memory	12 GB
Processor	Intel® Core™ i5-6200U CPU 2.30GHz x 4
Graphic Card	Intel® HD Graphics 520 (Skylake GT2)
HDD	64 GB

4.3.3. Image Transformation Phases Evaluation

The first set of experiments was based on verifying the variation of the precision when DCT is applied with different block sizes. In addition, in one of the aforementioned experiments, DWT is applied before the LBP execution. The CASIA v1.0 and CASIA v2.0 datasets were used for this purpose. In Table 4.3, the results obtained are shown.

Table 4.3: Variation of precision using DCT with different block size 8 × 8 and 16 × 16

Dataset	DWT-LBP-DCT		LBP-DCT	
	Block Size 8 × 8	Block Size 16 × 16	Block Size 8 × 8	Block Size 16 × 16
CASIA v1.0	77.86%	69.27%	85.71%	76.12%
CASIA v2.0	65.11%	65.83%	97.12%	74.10%

According to the obtained results, shown in the table above, the greatest accuracy is reached by using a DCT block size of 8 × 8. From here, all the experiments will be executed with this block size and will define the subsequent algorithm.

The second group of experiments is intended to define which chrominance channel shows the best results. The results obtained are shown in Table 4.4.

Table 4.4: Algorithm executed individually on each component Cr and Cb and also combined

Dataset	Cb	Cb + Cr	Cr
CASIA v1.0	98.57%	98.22%	99.64%
CASIA v2.0	99.50%	99.64%	99.64%
IFS-TC	96.19%	93.57%	93.57%
Average	98.10%	97.14%	97.62%

Since several previous works have compared the accuracy obtained between each chrominance channel and the combination of both, it is considered necessary to compare which channel suits the proposed algorithm with the best precision.

The obtained results have shown that the best option, for the proposed algorithm, is to use the Cb channel with an Average Accuracy of 98.10%, as this channel is the one that reaches the highest precision of all the used datasets.

Comparing both previous results, Table 4.5 shows the best results on each case with the above-defined block size.

Table 4.5: Comparison of the best accuracy of LBP-DCT with DWT-LBP-DCT only with Cb channel

Dataset	DCT Block Size 8 × 8	
	DWT-LBP-DCT	LBP-DCT
CASIA v1.0	85.71%	98.57%
CASIA v2.0	97.12%	99.50%

The third set of experiments was based on verifying the variation of the precision when applying QMF on the manipulation identification algorithm based on DWT.

The test was carried out by directly applying LBP on the Wavelet coefficients and alternatively by treating the wavelet coefficients with QMF before applying LBP.

The CASIA v1.0 and CASIA v2.0 datasets were used for this purpose. Table 4.6 shows the results obtained.

Table 4.6: Variation of the precisions when applying Quadrature Mirror Filter

Dataset	DWT-LBP-DCT	DWT-QMF-LBP-DCT
CASIA v1.0	97.66%	98.57%
CASIA v2.0	98.73%	99.50%

As can be seen in the table, the algorithm that uses QMF achieves a slight increase in accuracy in both data sets.

The fourth group of experiments was the execution of the proposed algorithm with all the acquired adjustments which increase the precision and the efficiency. The results obtained are shown in Table 4.7.

Table 4.7: Accuracy obtained by both algorithms

Dataset	DWT-QMF-LBP-DCT	LBP-Histogram-DCT
CASIA v1.0	98.57%	53.72%
CASIA v2.0	99.50%	94.94%
IFS-TC	96.19%	70.22%

As can be seen in the table, the algorithm based on DWT obtains better results in the three data sets and is faster than the algorithm based only on DCT. This is due to the overlapping blocks division that it performs. As a reference, the execution time of both algorithms was measured for an image with dimensions 1280 × 854. The algorithm based on DCT had an execution time of 89 seconds and the algorithm based on DWT, being more efficient, had a run time of 16 seconds. Likewise, it was observed that with a larger image size, the algorithm based on DCT considerably increases the processing time of the images. However, in the algorithm based on DWT, there is a slight increase in the execution time.

Finally, both algorithms get good results with compressed images, such as images with JPEG format. Good results were also obtained for images with PNG format. The two algorithms cannot reach a precision higher than 50% with images in Tagged Image File Format (TIFF) format.

4.3.4. Comparison of Other Techniques on the State-of-the-Art

Before drawing a general comparison of the proposed algorithm to others in literature, the results obtained from the test carried out over the different chrominance channels are compared with those of some other authors. The results of its algorithms are separately presented for each chrominance channel. Table 4.8 shows the obtained results from the proposed algorithm and the results from previous works.

Table 4.8: Chrominance channels' comparison

Approach		**Cr**		**Cb**		**Cb+Cr**	
Ref.	**Based on**	**CASIA v1.0**	**CASIA v2.0**	**CASIA v1.0**	**CASIA v2.0**	**CASIA v1.0**	**CASIA v2.0**
[ZLLW11]	RLRN	94.7%	-	94.3%	-	-	-
[AHA$^+$13]	DCT + LBP	95.80%	95.80%	96.50%	96.50%	97%	97.50%
[HSA$^+$14]	multi-WLD	92.62%	-	88.66%	-	94.19%	-
[KG16]	DWT + LBP	92.62%	94.09%	-	-	-	-
[HPME16]	Markov	-	-	-	-	97.92%	94.50%
[AH17]	DCT + LBP	96.52%	97.41%	96.19%	97.50%	96.90%	97.50%
[SE18]	Multi-Scale LBP+DCT	94.76%	97.30%	95.93%	9673%	-	-
[MJN19]	DCT + SVD	99.36%	-	98.46%	-	87.03%	-
[WSS19]	SIFT + SURF	90.5%		53%		99.0%	
Proposed	DWT+QMF+LBP+DCT	99.64%	99.64%	98.57%	99.50%	98.22%	99.64%

As the table shows, the proposed algorithm performs a better detection in all the tested datasets with all the chrominance channels, compared to the state-of-the-art algorithms.

Table 4.9 presents a comparison between the proposed authentication technique and state-of-the-art forgery detection methods, which use the same datasets and SVM classifier with RBF kernel. It can be observed that the proposed technique outperforms the existing techniques.

Table 4.9: Comparison between proposed technique and state-of-the-art approaches

Approach			**Type**	**Accuracy (%)**			
Ref.	**Based on**	**Color Space**	**Tampering Detected**	**CASIA v1.0**	**CASIA v2.0**	**DVMM v1**	**IFS-TC**
[SCC07]	MBDCT+DWT	RGB	Splicing			91.40%	
[ZKR08]	MBDCT+DWT+IQM	RGB	Splicing			87.10%	
[WDT10]	Markov	Cr	Splicing		95.6%		
[ZLLW11]	RLRN	Cr	Splicing	94.7%		85.0%	
[ZZ12]	MBDCT+LBP	RGB	Splicing			89.93%	
[ZWLL15]	DMWT+Markov	RGB	Splicing			93.36%	90.1%
[KG16]	DWT + LBP	Cr	Splicing	92.62%	94.09%	87.05%	
[HPME16]	Markov	YCbcr	Copy-Move Splicing	97.92%	94.50%	91.98%	
[AH17]	DCT-LBT	Cb+Cr	Copy-Move Splicing	97.0%	97.77% 97.50%		
[SE18]	Multi-Scale LBP+DCT	Splicing	Cr	94.76%	97.30%		
[MJN19]	DCT + SVD	Cr	Splicing	99.36%		96.52	
[AELTZ20]	CNN+DWT	-	Splicing	95.45%	97.27		
Proposed	DWT+QMF+LBP+DCT	Cr/Cb/Cb+Cr	Splicing Copy-Move	98.57%	99.50%	93.89%	96.19%

4.4. Summary

Digital images contain a large amount of relevant information. Because of this, they have become very important in legal procedures as they can be used as evidence of great value in a trial' resolution. For the images to be valid as evidence, it must be possible to guarantee their authenticity and integrity in a reliable way. Nowadays, there are numerous image editing applications that deliver highly professional results, thus making it harder to detect forgeries. Therefore, it is of great interest to have forensic tools for image forgeries detection that can guarantee the image integrity.

In this chapter a new approach for forgery detection that improves the results obtained in previous investigations was presented. This technique is based on the DWT with LBP histograms and QMF. This technique extracts the wavelet characteristics and obtains the histogram by applying LBP on the image under analysis. Finally, it improves the detection accuracy with QMF and classifies the image.

A set of experiments have been carried out using four benchmark datasets (CASIA v1.0, CASIA v2.0, DVMMv1 and IFS-TC datasets) of the state-of-the-art methods to evaluate the proposed technique. The results, after carrying out all of the experiments, show that the proposed algorithm achieves greater accuracy compared with the state-of-the-art algorithms using the same datasets and therefore provides more accuracy on tampering detection in most of the cases.

At the same time, the algorithm proved to be efficient in tests made after comparing their execution time with other algorithms.

Chapter 5

Techniques Based on JPEG Compression

This chapter presents two passive algorithms to detect forgeries in images with JPEG compression. The first one is an enhanced Error Level Analysis (ELA) technique, which can be used as an initial filter to detect the presence of splicing in an image. The second algorithm is a digital image authentication method based on the quadratic mean error of the CFA interpolation pattern estimated from the analysed image. The chapter is divided into five sections. Section 5.1 presents an overview of the methods used. Sections 5.2 and 5.3 specify in detail both proposed techniques, ELA and CFA respectively. A set of experiments to evaluate the effectiveness of the developed techniques in this chapter is described in Section 5.4. Finally, Section 5.5 summarises what has been presented in the chapter.

5.1. Overview

Passive tamper detection techniques look for specific artefacts which are created in the image processing pipeline within the camera or outside of it when any post-processing, that manipulates the image, is performed, as explained in Chapter 2. Hardware and software operations performed during this process are specifics to each camera and leave a sign on the final image. The proposed techniques in this chapter analyse the influence of traces left by JPEG compression and demosaicing algorithm in the forgery detection process. An overview of the artefacts generated in each of these processes is presented below.

5.1.1. JPEG Compression in Image Forgery Detection

The image compression process is made up of 4 phases: In the first phase, the raw data is divided into 8 × 8 blocks. The resulting blocks are reduced by performing colour transformation and down sampling processes in the second phase. In the third phase, DCT is used to transform the resulting data into the frequency domain. Later, the obtained DCT coefficients are zigzag ordered and quantified using the quantification table. Finally, in the last phase, a compression on the DCT coefficient is performed using loss-less entropy encoding to create the new JPEG file. Similarly, decompression of an image in JPEG format occurs by executing the reverse process. The final size of the JPEG image depends on the quality and content of the original image [CSE00]. Figure 5.4 graphically shows these processes.

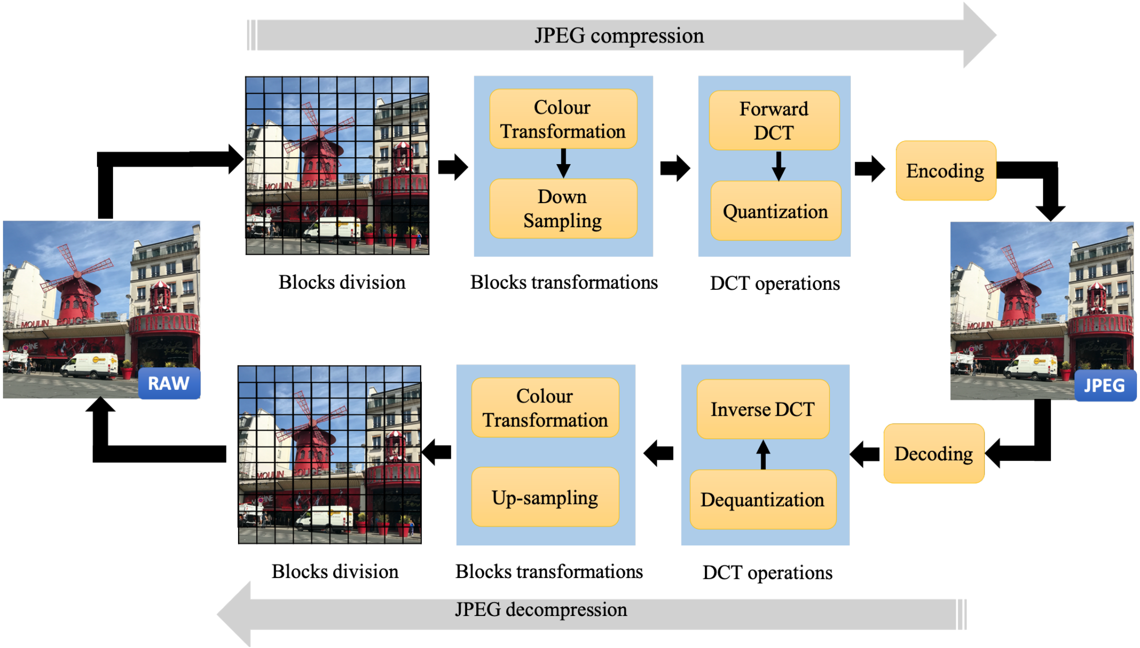


Figure 5.1: JPEG compression and decompression process

When an image in JPEG format is modified, a new JPEG image is generated. In the third phase of the JPEG compression process, encoding artefacts are generated that leave traces of manipulation. These traces can later be used to identify a tampered image.

5.2. Enhanced Error Level Analysis Detection Technique

The ELA algorithm focuses on the identification of areas with different levels of compression within the same image. A compressed image in JPEG format must have approximately the same level of compression in all its content. If there is an area with a significantly different error level, then there is a high probability that it has undergone a digital manipulation.

It could be said that ELA highlights the areas of the image most likely to degrade their colours in the next re-compression because the edited areas have a greater potential for degradation compared to the rest of the image.

The algorithm JPEG operates on an 8 × 8 pixel array, and each 8 × 8 square is compressed independently. If the image is not completely modified, all 8 × 8 squares should have similar error potentials, in other words, that when re-compressed each square will degrade at approximately the same speed. ELA re-compresses the image at a specific quality level. This re-saved image therefore introduces a known amount of error throughout the image, comparable to that of the original image. If the image is modified, each affected 8×8 square should have a higher error potential than the rest of the image, so the modified areas will appear with a higher level of potential error.

Our proposed algorithm aims to detect areas of the image that do not belong to the original content. It is developed in Python 3.7, using libraries specialised in image processing such as OpenCV and PIL. The input and output of the program is an image.

According to the ELA technique, explained above, an original JPEG image should have the same level of compression throughout its entire content. When the image contains a region that does not belong to its original content, the borders and textures of that area will be highlighted from the rest. Also, taking into account that the compression JPEG is adjustable we can know the compression level of the image content. Figure 5.5 shows the work flow of the proposed algorithm.

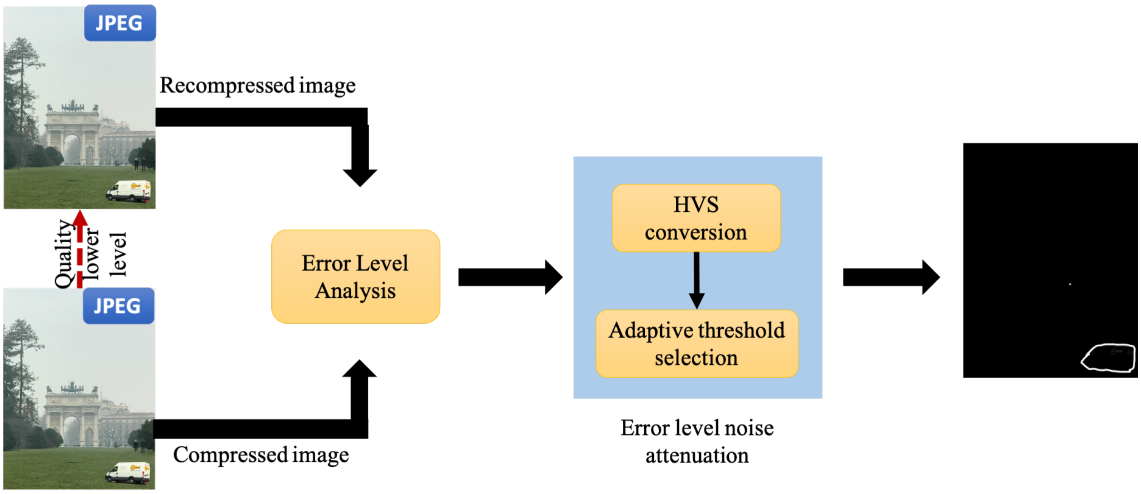


Figure 5.2: Proposed algorithm work flow

The algorithm needs two input parameters:

1. **JPEG image (I)**: A directory containing one or more JPEG images to analyse. For optimal results, it would be desirable that these images have the highest possible resolution.
2. **Quality lower level (QLL)**: An integer between 0 and 100 that represents the new level of JPEG compression of the input image. The most recommended is between 85 and 95.

For each image, a thread is launched that will treat the image and generate its output. The steps to detect splicing are the following.

The first step to analyse the image I is to re-compress it in JPEG format (I_{QLL}) with the quality lower level (QLL) declared as an input parameter.

Then, the absolute difference is computed between the image I and its re-compression version I_{QLL}). This operation is performed pixel by pixel for each RGB colour channel according to Equation 5.1.

$$ELA(x, y) = \frac{1}{3} \sum_{i=1}^{3} |I(x, y) - I_{QLL}(x, y)| \tag{5.1}$$

where x and y are the row and column for analysed pixel and i is each colour channel.

This difference is considered as the error levels associated with the original pixels. If the difference is large, then this pixel probably belongs to another image. With this result, it is possible to identify which pixel area has undergone a more significant change when applying the compression level recovering the maximum and minimum values obtained.

The values of the pixels that make up the resulting image are computed based on the maximum and minimum values previously calculated. To do this, they are scaled based on the 255.0 RGB value, and the brightness of each is enhanced.

After the above steps, a mask is applied to the generated image to highlight all the areas that have been left with blue and red tones, which are brighter than the rest of the content. As the mask covers the less bright areas, the RGB image is converted to the Hue-Saturation-Value (HSV) colour model. This conversion is carried out as working with HSV values makes it easier to isolate colours.

The conversion from RGB to HSV is governed by Equations 5.2 to 5.7.

$$R' = \frac{R}{255.0}, \quad G' = \frac{G}{255.0}, \quad B' = \frac{B}{255.0} \tag{5.2}$$

$$\begin{aligned} C_{max} &= max(R', G', B') \\ C_{min} &= min(R', G', B') \end{aligned} \tag{5.3}$$

$$\delta = C_{max} - C_{min} \tag{5.4}$$

$$H = \begin{cases} 60 \circ \frac{G'-B'}{\delta} mod 6 & C_{max} = R' \\ 60 \circ \frac{B'-R'}{\delta} + 2 & C_{max} = G' \\ 60 \circ \frac{R'-G'}{\delta} + 4 & C_{max} = B' \end{cases} \tag{5.5}$$

$$S = \begin{cases} 0 & C_{max} = 0 \\ \frac{\delta}{C_{max}} & C_{max} \neq 0 \end{cases} \tag{5.6}$$

$$V = C_{max} \tag{5.7}$$

In the colour representation HSV, the hue determines the colour you want, the saturation determines how intense the colour is, and the value determines the clarity of the image. To isolate colours, multiple masks must be applied, a low threshold mask and a high threshold mask for hue, saturation and value. Any pixels within these thresholds will be set to 1 and the remaining pixels will be zero. These thresholds values are calculated by applying the Adaptive Thresholding $T(x, y)$ computing weighted sum of block $B \times B$ neighbourhood of (x, y) where weights are computed using Gaussian window function $w[n]$ according to Equation 5.8 presented in [KKI17].

$$w[n] = e^{-\frac{1}{2}\left(\frac{n-\frac{N-1}{2}}{\frac{\sigma(N-1)}{2}}\right)^2}, \quad \sigma \leq 0.5 \tag{5.8}$$

where N represents the width, in samples, of a discrete-time, symmetrical window function $w[n]$, $0 \leq n \leq N-1$.

Once the mask has been applied, the output image of the program is left with the areas affected by splice marked with a white colour that stands out from the rest of the content which is either a black area or areas with white pixels, but it is isolated.

The image is saved in the output directory so that the researcher can check those areas which stand out most and compare them with the input image to verify whether or not that area belongs to the original content.

5.3. Colour Filter Array Based Detection Technique

The first step of this technique is to estimate the interpolation pattern of the colour filter matrix of the digital camera that captured the image. For this process the image is re-interpolated with various CFA patterns. For each pattern we get its Mean Square Error (MSE) between the original image and the re-interpolated image.

The results obtained from the MSE are then analysed to determine whether the image has been modified. It is expected that one of the values of the calculated MSE for each CFA standard will be much smaller than the other three. If none of the four values is significantly smaller than the others, it can be inferred that the image may have been post-processed. However, at this point you cannot be sure what type of modification has been made or whether it has been altered.

Being $L_c(x,y)$ the intensity of the colour channel image c in a spatial location (x,y) y $c \in \{R,G,B\}$, the next step is to define the colour filter mask that is done as shown in Equation 5.9.

$$\theta_{k,c}(x,y) = \begin{cases} 1, \ (x,y) \in \psi_{k,c} \\ 0, \text{ other case} \end{cases} \tag{5.9}$$

where, $\psi_{k,c}$ represents the location of the array of channel colour filters set c for a particular type of CFA pattern denoted by k and $\theta_{k,c}.(x,y)$ the corresponding colour filter mask of $\psi_{k,c}$.

The technique uses blocks of size $W \times W$, where $W = 8$ pixels, to divide the image taking into account only non-smooth blocks. Each non-smooth block is denoted as B_i where $i = 1, ..., N.$, being N the number of non-smooth blocks contained in the image. Blocks re-interpolated with the k filter are denoted as $\hat{B}_{i,k}$. These blocks are calculated by a convolution between the bilinear kernel and the re-displayed block B_i with the kth CFA pattern defined with Equation 5.10.

$$\hat{B}_{i,k} = f(B_i, \theta_k) \qquad k = 1, ..., 4 \tag{5.10}$$

Next, the MSE error between the blocks of B and $\hat{B}$ in non-smooth regions throughout the entire image is calculated using Equation 5.11.

$$E_i(k,c) = \frac{1}{W \times W} \sum_{x=1}^{W} \sum_{y=1}^{W} (B_i(x,y,c) - \hat{B}_{i,k}(x,y,c))^2 \tag{5.11}$$

where, E_i is an array containing the average quadratic errors for each colour channel.

To detect the relative distances between the colour channels a new error matrix E_i^2 is created. The normalisation of all rows of the E_i array is done with Equation 5.12.

$$E_i^{(2)}(k,c) = 100 \times \frac{E_i(k,c)}{\sum_{l=1}^{3} E_i(k,l)}, c = 1, \cdots, 3 \tag{5.12}$$

Because there are fewer pixels interpolated in the green channel, the values of the green channel column $V_i(k)$ are taken to determine if there is any modification. This process is done with Equation 5.13.

$$V_i(k) = 100 \times \frac{E_i^{(2)}(k,2)}{\sum_{l=1}^{4} E_i^{(2)}(l,2)} \tag{5.13}$$

By means of the uniformity of the vector V_i a possible post-processing operation can be indicated. The uniformity of the green channel vector is defined in Equation 5.14.

$$U(i) = \sum_{l=1}^{4} |V_i(l) - 25| \tag{5.14}$$

Finally, the median of the U vector is calculated as a CFA filter tracking metric as shown in Equation 5.15.

$$F = median(U) \tag{5.15}$$

The higher the CFA filter metric (F), the more likely it is that the image can be interpolated with the CFA filter. Therefore, it can be inferred that no significant processing or alteration occurred. Another way to measure the artifacts of the CFA chromatic interpolation algorithm is to observe the changes in the power of the sensor noise in the given image. If an image has been interpolated, the sensor noise in the interpolated pixels is expected to be suppressed. This is due to the nature of the low-pass interpolation. The variance of sensor noise in interpolated pixels becomes significantly lower than the noise power of the sensor in non-interpolated pixels. Therefore, the artifacts of the interpolation algorithm can be measured by comparing the ratio of noise variances of interpolated and non-interpolated pixels. If this ratio is close to 1, it can be assumed that the input image has been manipulated.

A typical way to obtain sensor noise is through the wavelet-based noise elimination algorithm presented in [DM09, SOGVAG+15]. This process is done on the green channel of an image by separating the interpolated pixels from the non-interpolated pixels using the green channel filter mask $\theta_{k,c}$, where $k = 1$ and $c = 2$.

The non-interpolated pixels are divided into 2 vectors A_1 and A_2 to obtain the ratio of the variations of the sensor noise to Equation 5.16.

$$F_2 = max(\frac{var(A_1)}{var(A_2)}, \frac{var(A_2)}{var(A_1)}) \tag{5.16}$$

where var represents the variance of the vector and max returns the highest value between x and y.

5.4. Experimental Results

In order to evaluate the effectiveness and performance of the proposed algorithms, several experiments have been executed. First, the parameters used in each algorithm are analysed to find which are most appropriate and later, the proposed algorithms are compared with other existing approaches in state-of-the-art techniques.

5.4.1. Datasets

For the evaluation of both algorithms, the public dataset CASIA v1.0 [DWT13] has been used. This dataset contains images manipulated by cropping and pasting operations using Adobe Photoshop CS3 version 10.0.1 in Windows XP. The spliced regions are from the same authentic image (copy-move) or from another image (splice). This is why only those dataset images containing the spliced region from a different image will participate in this experiment, as the algorithm is designed for splice detection. A specific dataset has also been generated for this experiment with high resolution manually spliced images. Table 5.1 shows a summary of the characteristics of the dataset used in the experiment.

Table 5.1: Datasets features

Datasets	Format	Resolution	Number of Images
CASIA v1.0 [DWT13]	JPEG	384x256	921 (splicing: 451)
Own	JPEG	1080x1920	30

5.4.2. Experiments Setup

The characteristics of the equipment with which the experiments were carried out are presented in Table 5.2. This is an important factor to bear in mind since the execution times of the different tests vary according to the computational resources available.

Table 5.2: JPEG compression quality low level evaluation

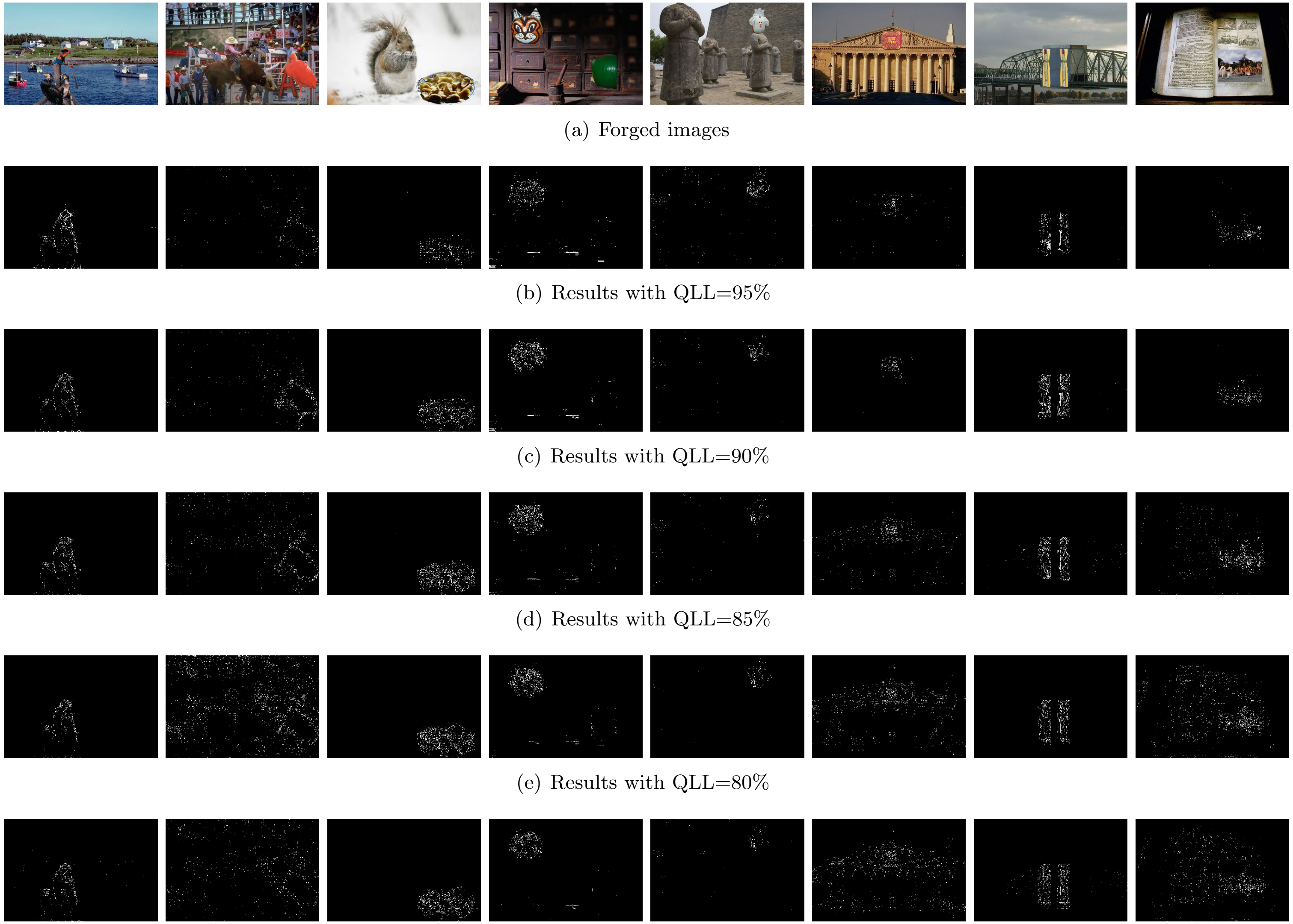
Resources	Features
Operating System	Ubuntu 18.04
Memory	4 GB
Process	Intel® Core^{TM} 2 Quad CPU Q8200 @ 2.33GHz x 4
Graphics	NV96
HDD	100 GB

5.4.3. Error Level Analysis Enhanced Technique Evaluation

The first set of experiments analyses the variation of quality levels in JPEG compression to find the quality low level appropriate to execute our ELA based algorithm. For this purpose, ten images forged with splicing were saved with different quality levels (75%, 80%, 85%, 90% and 95%) to observe how MSE and Peak Signal to Noise Ratio (PSNR) values are affected. The results are shown in Table 5.3. This experiments analyses the variation of quality levels in JPEG compression to find the quality low level appropriate to execute our ELA based algorithm. From the table, it is shown that using a quality level of 90% or 95% is better for JPEG compression algorithm. Therefore, we decided to use a quality level greater than or equal to 90% for the next experiments. Analogously it is observed in Figure 5.3 that the noise increases considerably from 85% of re-compression quality. However, the results obtained with our algorithm allow us to detect the splice-like manipulation in the images.

Table 5.3: Characteristics of the experimental equipment

Image	**95%**		**90%**		**85%**		**80%**		**75%**	
	PSNR	**MSE**	**PSNR**	**MSE**	**PSNR**	**MSE**	**PSNR**	**MSE**	**PSNR**	**MSE**
Img1	41.45	4.66	38.32	9.57	38.04	10.21	35.99	16.38	33.68	27.84
Img2	42.47	3.68	40.5	5.8	36.56	14.35	33.71	27.65	32.21	39.13
Img3	44.91	2.1	42.02	4.08	39.93	6.61	38.65	8.86	37.24	12.28
Img4	43.93	2.63	41.48	4.63	39.23	7.77	38.79	8.59	36.67	14.01
Img5	48.79	0.86	45.77	1.72	42.77	3.43	43.47	2.92	39.96	6.56
Img6	43.87	2.67	41.9	4.2	38.18	9.9	35.34	19.03	33.66	27.96
Img7	44.89	2.11	42.5	3.66	39.48	7.33	40.31	6.05	37.07	12.76
Img8	44.6	2.26	41.94	4.16	37.83	10.7	34.83	21.38	33.05	32.2



(a) Forged images

(b) Results with QLL=95%

(c) Results with QLL=90%

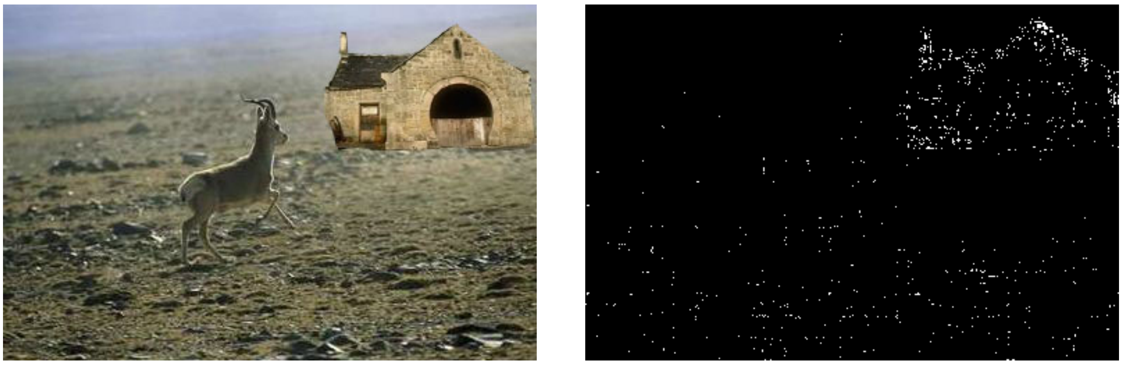
(d) Results with QLL=85%

(e) Results with QLL=80%

(f) Results with QLL=75%

Figure 5.3: Results of ELA algorithm varying quality low levels

The experiment carried out on this work was based on the verification of images which result from the application of the splice detection algorithm in digital images. This algorithm has been applied to the splice images of the CASIA v1.0 dataset for later revision. We have also used a dataset which has been specifically created for this experiment. The dataset consists of 30 high definition (1080 x 1920) images with splicing forgery within. The images were taken using mobile phones and then manually spliced using a specialized photo editing software to create a more realistic image and to deceive the human eye.The review consists of comparing with the naked eye if the regions that stand out the most in the resulting image are the regions that have suffered the splice. Figure 5.4 shows the positive results, the white area that stands out most in Figure 5.4(b) corresponds to the region pasted in Figure 5.4(a). The rest of the white pixels, being isolated, should not be taken into account. The Figure 5.5 shows a bad result of the algorithm, The white pixel zones in Figure 5.5(b) do not clearly distinguish which zone has been pasted in Figure 5.5(a). Table 5.4 shows the results obtained.



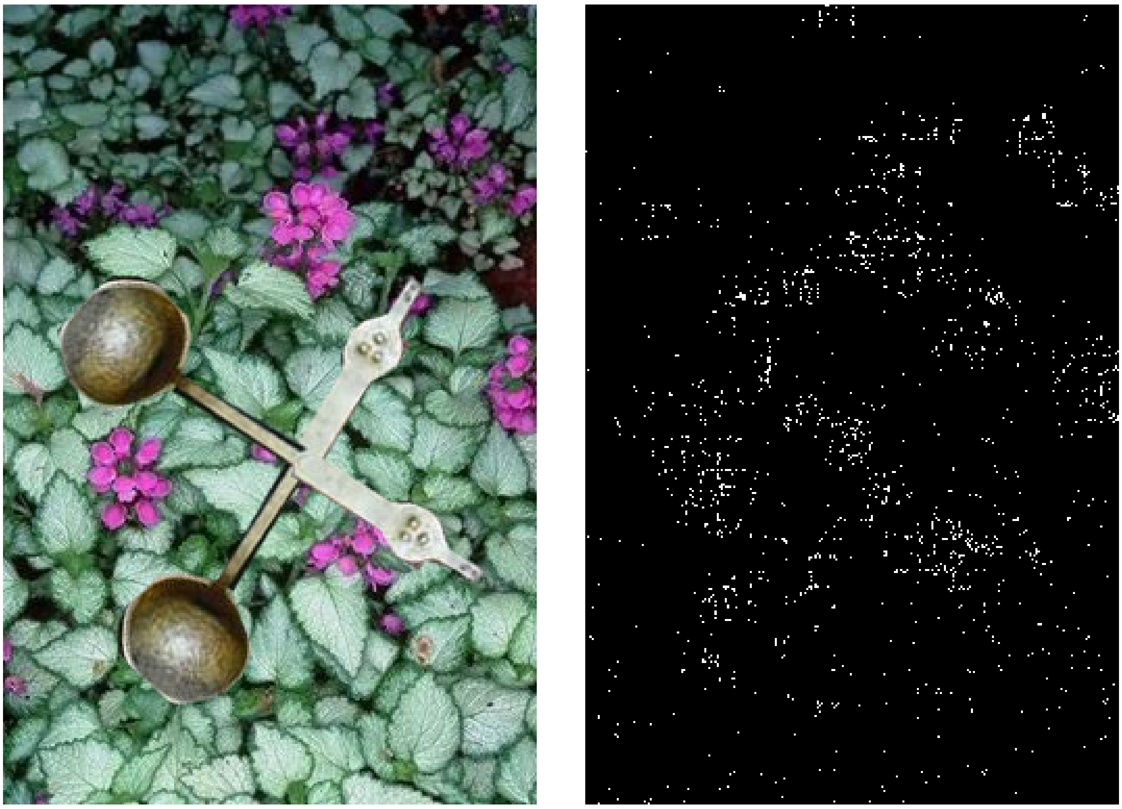
(a) Spliced image

(b) Result

Figure 5.4: Positive detection example on a spliced image

Table 5.4: Detection of positives after applying the algorithm

Datasets	**Number of Images**	**Positive**	**Performance**
CASIA v1.0	451	248 / 55%	00:00:25s
Own	30	22 / 73.3%	00:02:10s



(a) Spliced image

(b) Result

Figure 5.5: Negative detection example

5.4.4. Colour Filter Array Technique Evaluation

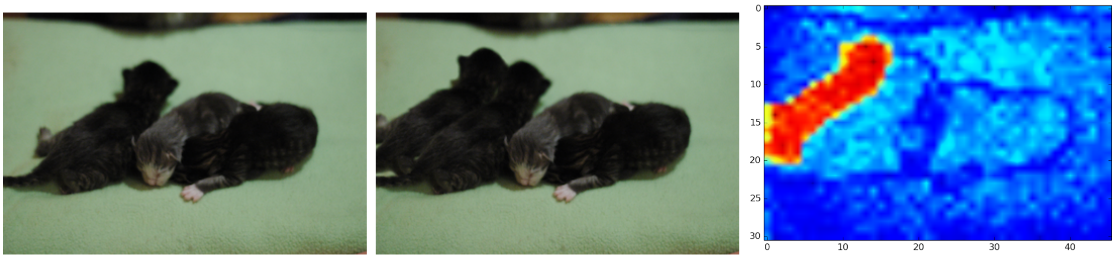
To evaluate the efficiency of the method described, we used the images of the datasets [CRJ+12a] and [ABM15], denominated D1 and D2 respectively.

D1 [CRJ+12a] dataset images have the following characteristics: High resolution images (3000x2000 or 2000x3000 pixels minimum), with realistic copy and move forgeries (”realistic” refers to the amount of pixels copied, the treatment of the pixels of the border of the copied region and the content of the region).

D2 [ABM15] dataset images have the following characteristics: The resolution of the images is of medium size (1000x700 or 700x1000), with uncompressed images with simply copied and moved regions. The uncompressed images depict simple scenes (an object, simple background) rather than complex scenes, since the dataset is used to study mainly the robustness against some specific attacks.

The characteristics of the equipment with which the experiments were carried out are presented in Table 5.2. This is an important factor to bear in mind since the execution times of the different tests vary according to the computational resources available.

To evaluate the efficiency of the described method, high resolution images (greater than 1500 × 1500 pixels) were used with and without alterations in different areas of the image [CRJ+12a] [ABM15]. In addition, the time it takes for the method to show the area where the image has been modified was measured. In the Figure 5.6 you can see the result obtained. Figure 5.6(a) shows the original image, Figure 5.6(b) shows the modification made and Figure 5.6(c) shows the result obtained when applying the proposed technique. It shows the region where the alteration was applied by highlighting the modified area. It should be noted that the dimensions of the image are 2000 × 3008.

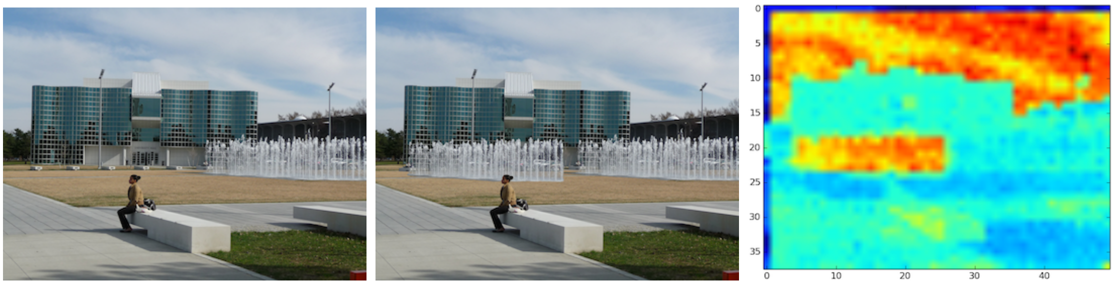


(a) Original image (b) Forged image (c) Forgery detection

Figure 5.6: Optimal results

However, there are cases where the results are not as clear due to the conditions of image, for example, when there are very clear backgrounds, such as skies, causing areas to be marked where there is no modification.

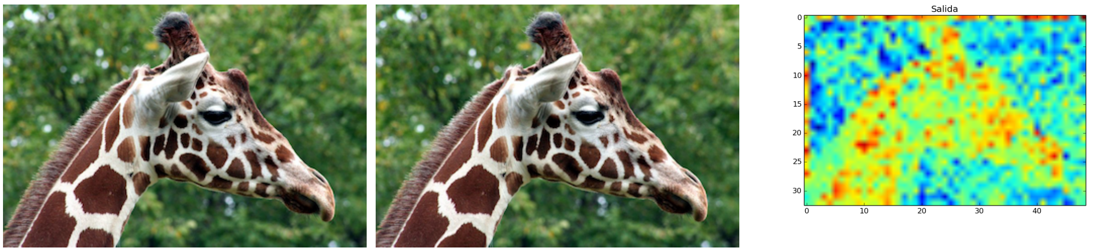
The Figure 5.7 shows this, although it does the delimitation of the modified area correctly, zones are shown in the part of the sky (Figure 5.7(c)) where calculations indicate that there is a forgery.



(a) Original image (b) Forged image (c) Forgery detected with false positives

Figure 5.7: Results with Errors

When analysing the results with small images from the D2 [ABM15] dataset (see Figure 5.8) it could be observed that the method is not accurate due to the low resolution of the image and that when processing the image and forming the blocks of size $W \times W$, described in previous sections, the lack of information in the image makes all the variances low and there is no significant difference between them.



(a) Original image (b) Forged image (c) Detection with false positives

Figure 5.8: Results over low resolution images

Table 5.5 shows the time it took for the method to analyse images of different resolutions. The time taken to process high resolution images was 24.2959 seconds which shows that the method is efficient and very accurate with large images.

Table 5.5: Method Execution Time

Resolution	2000x3008	2014x3038	2304x3072	2448x3264
Time (s)	20.3427	25.9153	22.6366	28.2889

The proposed method was developed in the Python programming language as it has libraries which facilitate image processing. The processing time of each image is low considering that all the information of the image is used without any previous processing and that the test images are of high resolution and are colour images.

5.5. Summary

The content of digital images possesses information that goes beyond the visual. This information is of great forensic value since its correct exploitation can guarantee the authenticity and integrity of the content. Because of this, digital images are an excellent source of evidence when it comes to resolving legal proceedings. The development and continuous improvement of new technologies enable conventional users to be able to alter the content of images and videos with professional results, invisible to the human eye. This is added to the fact that the detection of manipulations is a complex task and also requires continuous improvement to adapt to such a scenario, so it is essential to develop forensic tools capable of detecting these increasingly professional and common manipulations. The line of research that has been followed in this work begins with a study of the existing techniques of detection of manipulation on images dedicating more effort to techniques of detection of copy-move and splicing in images.

Two forgery detection techniques have been designed and developed: Firstly, an algorithm based on compression JPEG for splice detection in images. This algorithm uses the Error-level-Analysis technique and highlights those pixels with a different compression level and subsequently, areas that do not belong to the original content are marked in the image. Secondly, a technique to detect forgery in a colour image using chromatic interpolation algorithms. In the development of the work, it could be observed that by estimating the interpolation pattern and the mean quadratic error of blocks in the image, it can be determined whether or not there is a modification in a given image.

For the first detection technique, the CASIA v1.0 public dataset and a personal dataset have been used. The results show that they depend directly on the quality of the image because, for CASIA, the algorithm presents difficulties in detecting the region that does not belong to the original image. However, the dataset itself contains high-resolution images where 73.3% of accuracy was obtained. It is important to mention that the accuracy of the result is determined by the analyst since they are the person who has to consider whether the spliced area is clearly highlighted in the image obtained. Therefore, it is necessary to have thorough knowledge of how ELA works. However, this technique serves as the first piece of evidence, since, being a rapid response detection technique, it can serve as a source of suspicion for a researcher to further investigate those images that have presented suspicious white regions.

For the CFA algorithm, satisfactory results were obtained when entering images with dimensions greater than 1500×1500 pixels delimiting the modified area. Images that have a big area with white colours create false positives because the variance calculated in these sections is very low, compared to the rest of the image. Likewise, the best results are obtained with large images since the image information is sufficient to calculate the variance of the image correctly, and a distinction can be made between the variances. However, with images smaller than 700×700 pixels, the method has difficulties in detecting the area with modifications since the image information is not sufficient to make the difference between the variances known.

Chapter 6

Techniques by Estimating Interpolation Patterns

Several works have addressed the problem of detecting manipulations in images acquired from devices that use colour filter arrays, typical in the market due to low production costs. These devices use chromatic interpolation algorithms during the image formation process, allowing them to perform statistical analyses of inconsistencies generated from this process for authentication purposes. Most of the works focus on analysing the green band of the Bayer filter since it contains more information than the blue and red bands. The lack of methods for effectively analysing other bands or different colour filters reduces the detection capability of known tools. The main purpose of this work is to provide a general methodology for detecting manipulations in this type of devices, in addition to providing new techniques that allow generalising the analysis in a great diversity of sensors.

6.1. Technique Description

Derived from the revision of multiple proposals that deal with the identification of modified images through the analysis of periodic inconsistencies or artifacts, we present a methodology that consists of 5 phases:

1. **Image estimation:** The first step is to obtain an image estimate that minimises artifacts produced from post-processing processes, such as chromatic interpolation.

2. **Error calculation:** Subsequently, an error matrix is obtained. This consists of the difference between original and estimate images.

3. **Probability map:** Though the error matrix contains evidence that can be used to detect possible manipulations, the computation of probability maps helps to obtain better results by providing normalised values in the $[0, 1]$ range, which improves the filter pattern detection.

4. **Feature extraction:** The error matrix is partitioned into blocks of a predefined size for individual analysis and classification using statistical methods. The size of the block must be selected properly, as the analysis of very large blocks may ignore small manipulations, while small blocks lead to statistically inaccurate results, thus making correct detection difficult.

5. **Segmentation:** Finally, a decision is made on whether the image has been modified. The identification of the areas where the image has been modified depends to a great extent on the effectiveness of the previous 4 steps.

Each of the phases is detailed below, describing the tools used in each step. Some of these tools are generalised for its application in different colour filters.

6.1.1. Image re-Estimation

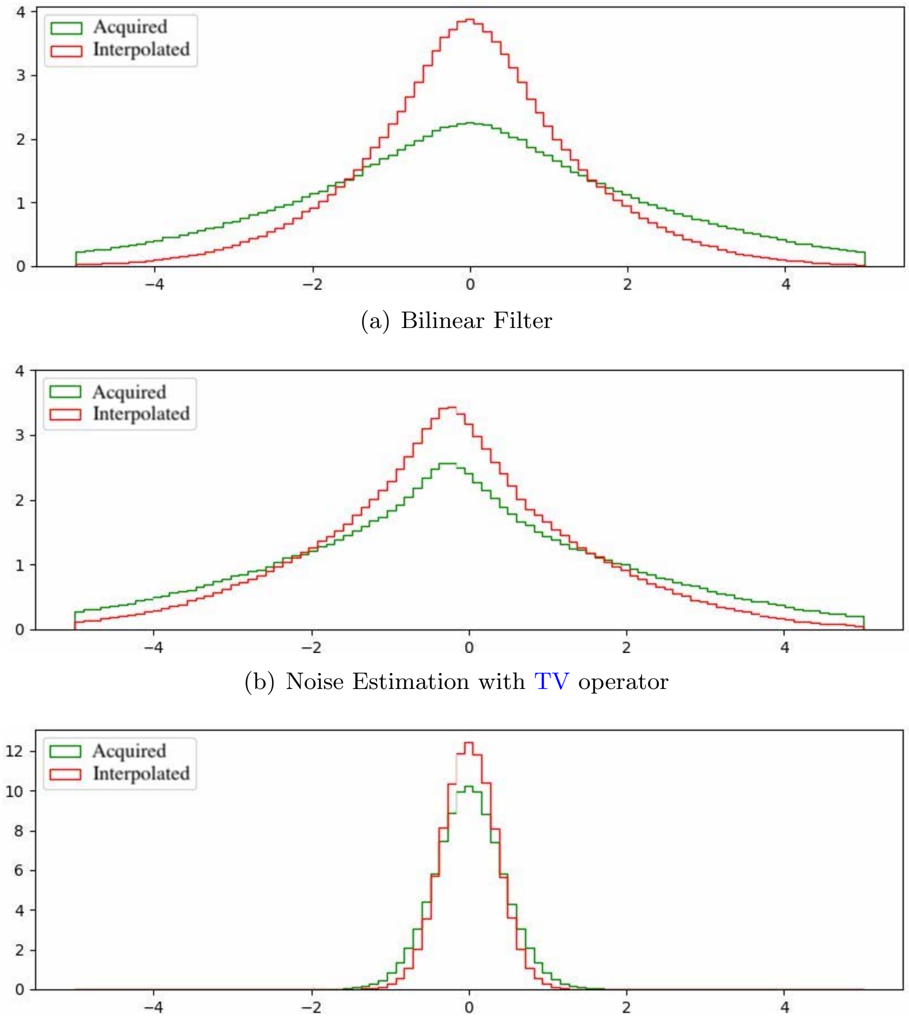
The first step is to estimate the image in such a way that the difference between the original and estimated images discloses the interpolation artifacts. The path followed in [PF05] and [FBDRP12] to carry out this step consists of estimating the coefficients of the chromatic interpolation model used by the source device to generate a new image. Although in [PF05] it is argued that an analogous procedure can be used for the other bands, this may be inaccurate due to the complexity of the algorithms used in real devices. A list of known algorithms is detailed in [MC11]. This causes most experiments and results to focus on the green band. Simplifying this task, in [GFSOGVHC18] an estimate using OLS only with the acquired pixels is considered. However, as mentioned in [DM09], and proved experimentally, simpler methods such as bilinear and bicubic interpolations make it possible to obtain good results, reducing the computational complexity.

An additional option detailed in [DM09] is to apply a denoising algorithm to the analysed image, in this case, by means of the DWT transformation. In general, precise noise estimation methods help to detect these artifacts since the noise is suppressed by the interpolation algorithms applied to the Bayer filter. In this work, results on denoised images using the Total Variation (TV) operator [Cha04] are also considered.

6.1.2. Error Estimation

Once the estimated image is obtained, the difference between the two images evidences the difference between the distributions of each class. Figure 6.1 shows the distribution of interpolated and acquired pixel errors by applying 3 different techniques in the image estimation phase: bilinear blurring, denoising with the TV operator, and denoising with the DWT transform.

In all cases, the difference in acquired and interpolated pixel distributions is evident, although to a greater extent in the case of bilinear blurring.



(a) Bilinear Filter

(b) Noise Estimation with TV operator

(c) Noise Estimation with DWT

Figure 6.1: Residuals obtained through different image estimation techniques

Due to the good results obtained with bilinear blurring and denoising techniques, methods of approximation of the interpolation kernel, which are computationally expensive with an almost negligible increase in the efficiency of the analysis, have not been considered. As for the denoising algorithms, experimental tests have shown better performance in the case of the TV operator.

6.1.3. Probability Map

Following the idea proposed in [PF05], each pixel is assigned a level of membership to the set I. Originally, the errors are considered to follow a mixed probabilistic model, where the interpolated pixels have a normal distribution and the acquired pixels a uniform distribution in the range of intensity values (0 to 255 commonly). In [GFSOGVHC18] it is observed that using the Equation 6.1, based on the complementary error function, provides better results by discriminating acquired pixels more effectively.

$$f(x) = 2\left(1 - \Phi\left(\frac{|x|\sqrt{2}}{\sigma}\right)\right) \tag{6.1}$$

In order to apply existing methods to every colour band and colour filters other than the Bayer filter, the proposals can be adapted to each particular case. This introduces the restriction of knowing a priori the pattern of acquired and interpolated pixels. Although in many cases this is not a strong restriction, experimental results show that this approach does not work in most cases, mainly because artifacts are lost after complex demosaicing and post-processing methods. The pattern of the artifacts is estimated to achieve a more suitable method, generalising the process for use in various filters. This is potentially useful for models using less-known filters such as RGBW [Tac12] and X-trans [Fuj], size 4×4 and 6×6, respectively. This process is detailed below.

Consider a pair of positive integers (c_1, c_2), which represent the size of the colour filters (or multiples of it) are chosen. For each pair (u, v) with $1 \leq u \leq c_1$, $1 \leq v \leq c_2$ the following sets are considered

$$I_{u,v} = \{(i,j) \mid i \mod c_1 = u, j \mod c_2 = v\}$$

These sets represent the selection of a single pixel in disjoint blocks of the size of the colour filter $c_1 \times c_2$. To exemplify this, in the case of the Bayer filter, $c_1 = c_2 = 2$. The set of pixels acquired from the red band is provided by $I_{1,1}$, for the green band $I_{2,1} \bigcup I_{1,2}$ and finally for the blue band $I_{2,2}$. The estimated variance σ^2 required for the generation of the probability map is defined by Equation 6.2.

$$\sigma^2 = \min\{((\mathcal{I}_{ij}))|1 \leq i \leq c_1, 1 \leq j \leq c_2]\} \tag{6.2}$$

By computing the minimum variance between the possible configurations, it is expected to obtain a good estimate of the variance for the interpolated is expected to obtained.

In this work, an additional filter is applied to increase the presence of interpolation artifacts in the probability map P. This filter consists of calculating the average of the values in the m-nearest neighbours of the same class. This is formally defined in Eq. 6.3.

$$\hat{P}(i,j) = \frac{\sum_{(k,l)\in B_{ij}} P(k,l)}{|B_{ij}|} \tag{6.3}$$

where

$$B_{ij} = \{(k,l) \mid k = i \mod c_1, l = j \mod c_2, -m \leq k, l \leq m\}$$

Figure 6.2 shows the results obtained by applying 3 different probability maps in two areas of the image, one original and one tampered. The probability maps correspond to those proposed in [PF05, GFSOGVHC18] and the one proposed in this work. Probability maps of tampered (red) and original (green) zones of the image (a) computed using:

Gaussian Mixture Models (GMM) in (a) and (b), the function based in $erfc$ in (c) and (f) and the proposed method with the mean of the m-nearest neighbours in (d) y (g). We can see greater discrimination is seen of acquired pixels calculating the map using the Equation 6.3, proposed in this work, which allows for more precise identification of manipulations.

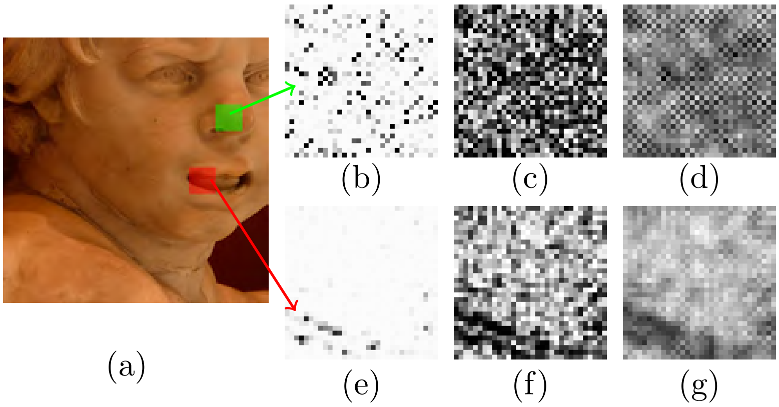


Figure 6.2: Image obtained by applying 3 different probability maps

6.1.4. Feature Extraction

The next step is to create a partition of the image in blocks of size $b \times b$ and from each block extract a value that will be analysed later by segmentation methods to determine the manipulated zones. In general, we can distinguish 2 techniques used in previous works:

1. **Comparison of the variance between acquired and interpolated pixels**: In [DM09] this is directly verified as the quotient of the variances of acquired and interpolated pixels. A more refined proposal is detailed in [FBDRP12], where a matrix of local variances is generated pixel by pixel, allowing analysis of small blocks.

2. **Frequency analysis**: These consist of verifying that artifacts occur periodically according to the Bayer filter pattern, as proposed in [PF05] and [GFSOGVHC18], where the magnitude of the DFT and DCT transformations are used accordingly.

Continuing with the purpose of this work, a new feature is proposed that can be applied to the probability map regardless of the band or colour filter. Additionally, it is observed that this approach allows the error matrix to be analysed directly, as long as periodical artifacts are present.

To approximate the shape of the periodic pattern, all disjoint blocks of size $b \times b$ are used, as shown in the Equation 6.4.

$$S(i,j) = \frac{1}{M/bN/b} \sum_{k=0}^{M/b} \sum_{l=0}^{N/b} P(bk+i, bl+j) \tag{6.4}$$

The matrix S of size (b, b) consists of an approximation of the filter pattern. The element $S(i, j)$ is the average of the elements in the position (i, j) in each block of size $b \times b$ extracted from the probability map. For the Bayer filter, a pattern with big values followed by smaller ones is obtained. For this reason, it is not necessary to know the precise distribution of the interpolated and acquired pixels a priori. Subsequently, the correlation of the average pattern S with each of the extracted blocks is calculated. Those with a higher correlation are considered authentic, while those with a low correlation will be labelled as altered. This feature is formally defined by Equation 6.5.

$$C = (P, S) \tag{6.5}$$

In Figure 6.3 the behaviour of the features extracted from 8×8 blocks of a manipulated image can be observed. The presence of a mixed distribution allows, in most cases, to effectively detect the area of the image that has been compromised. Using this feature, we have managed to reproduce the results obtained in [FBDRP12] and [GFSOGVHC18], where the green band is analysed. In addition to this, this method has been successfully applied to all the bands of the RGB and YUV colour spaces without further modification, which provides this method with great flexibility. These results are detailed in the Section 6.2.

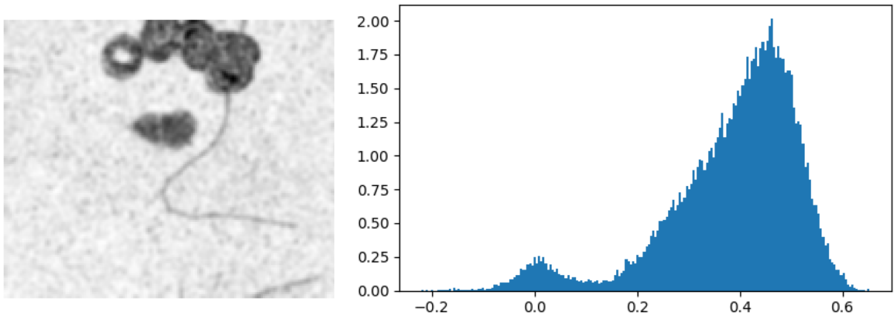


Figure 6.3: Distribution of the values of the proposed feature

6.1.5. Localizing Modifications

The extracted features are analysed using statistical methods to classify the corresponding blocks in two groups: original and modified. Two of the techniques commonly used for this purpose are introduced in the next section.

6.1.5.1. Gaussian Mixture Models

These are probability models in which the distribution of the elements of the total population can be explained by the behaviour of certain sub-populations. In the case of GMM, the population is considered as consisting of two subsets that follow a normal distribution with different parameters. In the case of features considered in this work, the populations are the values obtained from manipulated and original blocks. The former case is expected to follow a normal distribution with mean μ_0 close to zero, due to the low correlation between original and modified blocks, while the mean for original blocks is much higher. The parameters of both distributions can be estimated using the EM [TM08] algorithm. Once the parameters for suspicious elements (μ_0, σ_0) and for the original elements (μ_1, σ_1) have been estimated, it is possible to propose the segmentation threshold as the value x such that:

$$\frac{\phi_{\mu_0,\sigma_0}(x)p_0}{\phi_{\mu_0,\sigma_0}(x)p_0 + \phi_{\mu_1,\sigma_1}(x)p_1} = 0.5 \tag{6.6}$$

where p_0 and p_1 are the a priori probabilities and are estimated according to the number of elements in each group.

6.1.5.2. Otsu's Segmentation

The Otsu's Segmentation method [Ots79] is a widely used algorithm for the segmentation of monochromatic images. Its goal is to find the point at which segmentation produces two groups whose combined variance is minimal. Formally, for a set S, the segmentation value s is searched in such a way that when defining a partition $S_0 \bigcup S_1$ of S as follows:

$$S_0 = \{x \in S \mid x \leq s\}, \quad S_1 = \{x \in S \mid x > s\}$$
$$p_i = \frac{|S_i|}{|S|}, \quad \sigma_i^2 = Var(S_i)$$

the value of σ^2 defined by Equation 6.7 is minimized.

$$\sigma^2 = p_0\sigma_0^2 + p_1\sigma_1^2 \tag{6.7}$$

6.1.5.3. Proposed Segmentation Method

Finally, a segmentation is proposed that allows a better location of the affected areas when the feature defined by Equation 6.5 is used. The matrix of extracted features is denoted by C the matrix of extracted features. The first step is to define an initial threshold t, close to 0. Examining the matrix C, input by input, those elements whose value is below this threshold are marked as suspicious. Subsequently, the threshold is increased by a small value, and for each element marked as suspicious, its m-nearest neighbours are marked in the same fashion. This process is iterated until stability is reached, that is, until there are no new blocks added, or until a predefined threshold value is reached.

The main idea of this technique is to expand the zones that have obviously been tampered with without adding new blocks, which visually do not provide information. A description of this procedure can be seen in Algorithm 2.

Algorithm 2: Segmentation with k-nearest neighbours

```
Input: $C$, $t$, $\epsilon > 0$ and $m$
Result: Segmentation $T$ of the feature matrix
$T \leftarrow \{(i,j) \mid C(i,j) < t\}$;
$T' \leftarrow \emptyset$;
while $T \neq T'$ do
    $t \leftarrow t + \epsilon$;
    $T' \leftarrow \tau T$;
    foreach $(i,j) \in T$ do
        foreach $(i',j') \in [i-m, i+m] \times [j-m, j+m]$ do
            if $(i',j') < t$ then
                $T \bigcup (i',j')$;
return $T$
```

Figure 6.4 shows the process of segmentation of a manipulated image *(a)* its ground truth *(c)*, the feature matrix *(b)* and the segmentation obtainer using Otsu's method *(d)*. Figure 6.4(e) shows a cleaner result in the area located by the proposed algorithm. Although it is possible to apply Otsu and GMM-based segmentation methods iteratively, the selection of the appropriate value can be complex and introduce noise by applying strict segmentation.

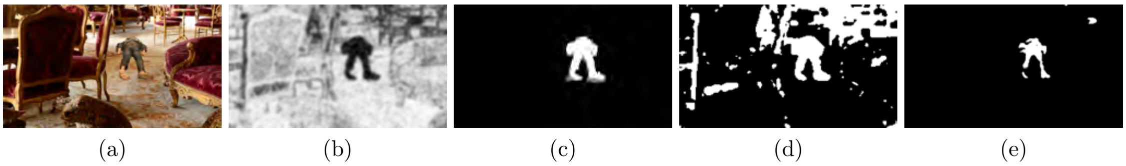

(a) (b) (c) (d) (e)

Figure 6.4: Segmentation of the features of the manipulated image

6.2. Experimental Results

Throughout this section, all the experiments that have been carried out to evaluate the effectiveness of the proposed technique will be shown.

6.2.1. Datasets

To evaluate the proposed method, two public datasets of manipulated images were used and can be found at [CRJ$^+$12b] and [KH17]. These datasets are denoted as D1 and D2 in the following sections. Publicly available datasets were chosen to avoid the creation of ad-hoc material, which can induce biased and unrealistic results.

- **RAISE [KH17]:** D1 consists of 4 groups of 55 images modified by each of the following devices: Nikon D7000, Nikon D90, Sony $\alpha57$ and Canon 60D. The manipulations of this dataset are diverse and are not detailed in the source.

- **Image Manipulation Dataset [CRJ$^+$12b]:** D2, in turn, has 4 groups of 12 images in which only copy and move modifications are applied. The groups of D2 correspond to a social network (Flickr) and 3 devices: Panasonic, Nikon and Canon, although the model is not specified.

6.2.2. Experiments Setup

The implementations have been done with the help of libraries dedicated to numerical analysis and images available for Python: Numpy, Matplotlib, Skimage and OpenCV. The performance of the algorithm was measured by contrasting the area obtained against the truly modified zone, indicated by masks provided in the same datasets. It was defined that a detection is correct if at least 5% of the modified zone is properly identified and can be distinguished in a precise way.

To analyse the modifications in both datasets, the following configuration of parameters has been used:

- For the estimation of the image the method of noise extraction based on the TV operator is used.

- A probability map is calculated using the membership function defined by the Equation 6.3 using $m = 3$ nearest neighbours.

- The extracted feature is defined by Equation 6.5 using blocks of size 8×8.

- Finally the segmentation is carried out by applying a Gauss filter on windows of size 3×3 and then using the Algorithm 2 with $t = 0.05$, $\epsilon = 0.05$ y $m = 2$.

6.2.3. Discussion of Results

Figure 6.5 shows the results from applying the proposed algorithm in images from different devices. In *(a)* the manipulated image is shown, in *(b)* the probability map calculated from the errors. In *(c)* the modified part exhibited by the features of the blocks is already observed. In *(d)* there is the manipulation map, which can be contrasted with the genuine, manipulated zone shown in *(e)*. In the first row, we see the analysis of the image shown above. In the second row it is possible to notice that the analysis can clearly identify manipulations.

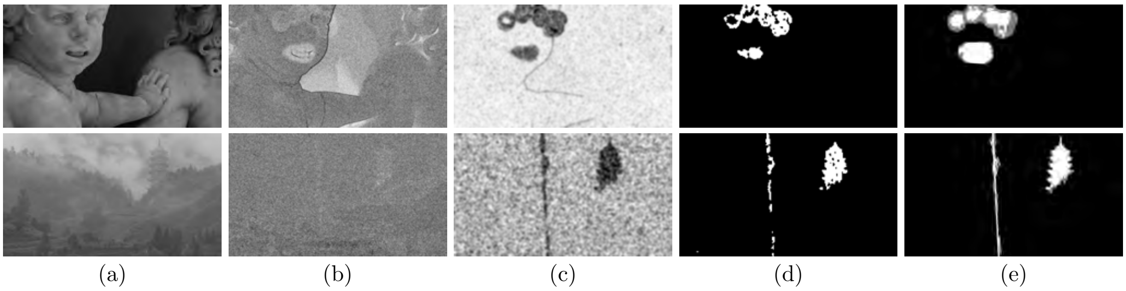


Figure 6.5: Results of the analysis in 2 images of different devices

In the Nikon models we have analysed the RGB bands separately and the Y-band of the YUV transformation have been analysed separately, obtaining similar results. The analysis for Sony $\alpha57$ has been much more successful using the U and V bands of the YUV transformation. In all cases, a great success rate was obtained to decide if an image has been manipulated. In the images belonging to Nikon devices it was possible to decide that 52 of the 55 images had been manipulated, while in those belonging to the Sony $\alpha57$ camera, this was achieved in 53 of the 55 images. Despite the good results, we found wide differences in the percentage of manipulation detected. This can be seen in detail in Table 6.1 in which the analysis in each dataset is presented: the source of the image in column 2, the number of images recognized as manipulated in the third column and the number in a percentage range of detection in columns 4 to 6.

Table 6.1: Detail of the analysis in each dataset

	Device	**Identified**	**% of forgery identified**		
			<25%	**25-50%**	**>50%**
Dataset 1	Nikon D7000	52/55	12	10	30
	Nikon D90	52/55	13	11	28
	Sony $\alpha57$	53/55	4	8	41
	Canon 60D	0/55	0	0	0
Dataset 2	Panasonic	10/12	0	0	10
	Nikon	9/12	0	1	8
	Canon	9/12	0	0	9
	Flickr	9/12	0	0	9

Continuing with dataset D2, Table 6.1 shows that the proposed method has been successful in most of the images from the Panasonic, Nikon and Canon cameras. In some elements of this set of images, we have observed an unexpected pattern in blocks of size 8×8, which consists of the presence of a pixel whose error is greater than the rest. When making a copy and move type modification, this element loses the relative position that was observed in the original area, rendering the manipulation detectable. This allows the performance of detections even without the use of the probability map, and directly into the estimated noise.

Finally, detections in the set of pictures from Flickr have also been successful in the majority of the provided images, even after uploading and downloading the image from the aforementioned social network. An analysis of metadata performed on other uploaded pictures reveals that no meaningful changes are carried out. Thus the accuracy can be explained because the original is, at least, not compressed with lossy algorithms. Figure 6.6 shows the results of performing the detailed technique on the green band of the RGB colour space and in the V band of the YUV colour space. In the first row, tampered images are shown. Second row shows the ground-truth. Third row shows results after analysing the green band, performed in previous works. The last row shows results in the V band of the YUV colour-space. It can be seen that the analysis made on the green band fails to detect some modifications, while the same analysis in the V band provides more accurate results. In the third column, it can be seen how segmentation can introduce difficulties for identifying the tampered zone accurately. However a visual inspection of the results helps to define it and handmade cleaning can be carried out in most of the cases.

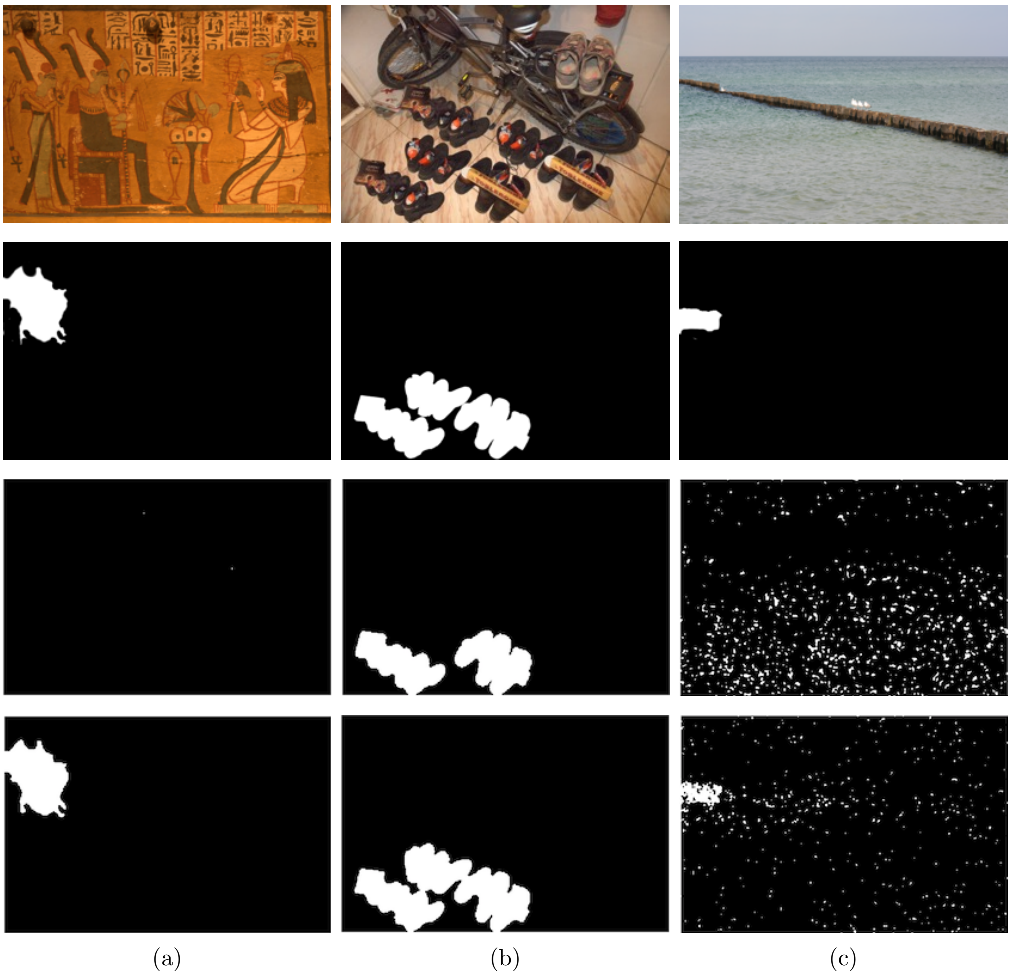


Figure 6.6: Results obtained from different colour spaces

Considering that similar techniques, briefly introduced in Chapter 3, are applied only on the green band, the results are similar to those obtained by the proposed analysis only when applied on the same band. The flexibility of the method allows performing the tamper detection technique in any colour filter, band and colour-space, resulting in successful detections even when the green pattern remains aligned after applying copy-move or splicing. However, in some cases, more expertise is required to select block size and neighbour parameters for the probability map. In Table 6.2, a fast comparison of related methods is shown. It can be noticed that the proposed technique can be performed using a small block size for localised and accurate detections, and can also be applied directly in other colour spaces and bands, outperforming known techniques.

Table 6.2: Comparison of the proposed technique with existing detection algorithms

	Tampering method				**Minimum**	Band
	Blur	**Copy-move**	**Postprocess**	**Splicing**	**block-size**	
Popescu	✓	53	✓	✓	256×256	G
Dirik	✓	53	✓	✓	96×96	G
Ferrara	✓	✓	✓	✓	2×2	G
González	✓	✓	✓	✓	2×2	G
Le	✓	✓	✓	✓	2×2	G
Proposal	✓	✓	✓	✓	2×2	RGB/YUV

6.3. Summary

A methodology has been proposed to generalise the detection of manipulations to different colour filters without losing precision in the results observed in the literature dealing with the RGB case. This method can be applied in different scenarios, as long as periodical artifacts are found in the analysed image. The inspection of every phase of the methodology proposed in Section 6.1 makes it clear that it is possible to consider other tools in each of the steps if an improvement of the results is pursued. Modifications can also be proposed to address particularities that might arise during the analysis of certain issues, such as known filters, post-processing or compression. In this sense, experimental results have shown that, in addition to the 2×2 pattern left by the demosaicing process, other patterns of a different size could appear as a result of the operations executed by the device in the image formation process.

Though the methodology here proposed can be successfully applied to a broader range of devices, the offered generalisation is far from working for every case, as it can be from several existing works addressing this issue. The arguments given in the last paragraph must be taken into account when studying a specific analysis. Different lines of action can be contemplated in each phase depending on the characteristics of the statistical evidence found, or also form information known a priori (device data, type of modification, compression rate, etcetera).

It has been observed that some of the problems involved in the definition of define the modified region come from the difficulty in defining adequate parameters in the segmentation methods, so a more careful analysis of this phase is necessary. As future work, the following objectives have been identified:

- Study the features that allow identifying the manipulations with more detail. Though some of the periodical patterns are well understood, as is the case of interpolation artifacts, some other patterns have arisen, which can be associated with compression or post-processing procedures. Understanding how these artifacts are created will help in improving detection techniques.

- Provide better segmentation algorithms suitable for the obtained probability maps. It has been noticed that these maps provide meaningful information to localise the tampered regions, which can be easily visualised. But at some cases, the segmentation introduces noisy regions which make it difficult to obtain more accurate observations.

Chapter 7

Technique Based on DCT Blocks Features

This section proposes a method for detecting Copy-Move type manipulations on an image. The proposed method uses the discrete cosine transform to extract features from its coefficients and thus construct transfer vectors which are grouped together. Then, using a tolerance threshold, it is determined whether or not there are regions that have been copied and moved within the analyzed image. The algorithm is based on the technique introduced by Fridrich [FSL03].

7.1. Technique Description

A diagram presenting the main processes of the detection algorithm can be found in Figure 7.1. Further details will be provided throughout the rest of the section.

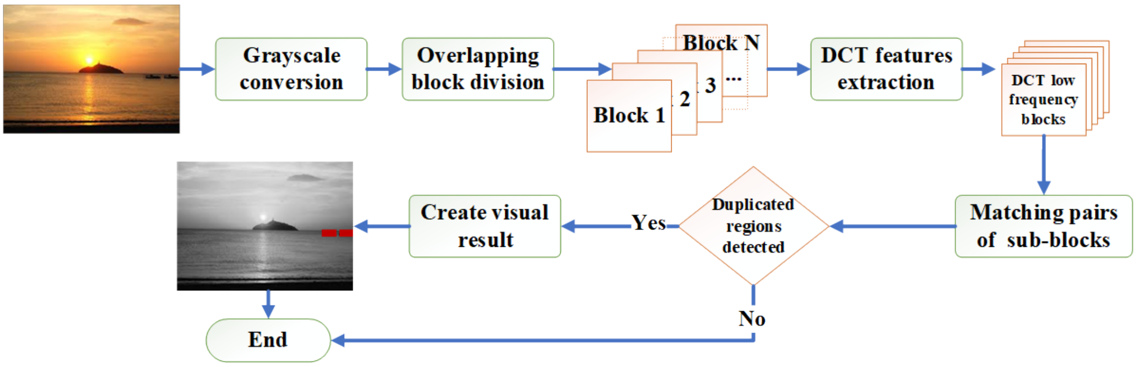


Figure 7.1: Processes of the Copy-Move detection algorithm

Summary detection algorithm is described below:

1. Transform the image to grayscale.

2. Divide the image into small overlapping blocks of size $B \times B$ from top-left to bottom-right.

3. Compute the DCT transformation of every block, sort the coefficients in a zig-zag fashion and truncate the list to contain the first k elements.

4. Lexicographically sort of the truncated coefficient lists, and for each list, compute a similitude measure between its nearest neighbours. If the similarity is lower than the threshold, blocks are considered as identical.

5. For every pair of identical blocks, the translation vector is computed. If the number of vectors in a given direction exceed a predefined quantity, every block is considered as part of the copy-move tampering.

The block size and thresholds chosen for the algorithm should be dependent mainly on the size of the image and the expected size of the modification. Some recommendations for the parameter selection will be given according to results obtained in the experiments detailed in Section 7.2. Now, each step of the aforementioned technique will be explored in detail.

The input parameters and the expected output after algorithm execution are the following:

- INPUT: Suspicious image I.

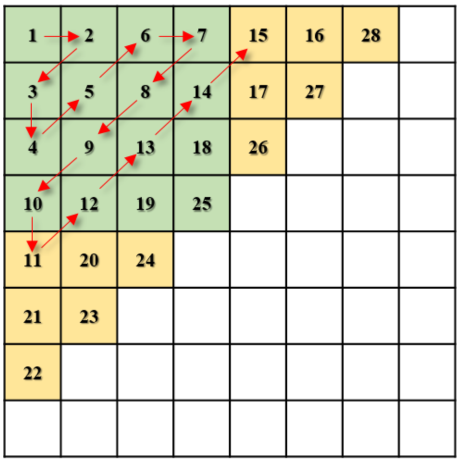
- OUTPUT: greyscale image with the original and probably tampered blocks painted in black.

First, images are commonly regarded as a combination of the red, green and blue (RGB) channels. Theses values are commonly provided by Bayer filters in many image devices. The grayscale image, also known as *luminance*, is obtained by combining the RGB components according to the linear transformation 7.1.

$$Y = 0.2125R + 0.7154G + 0.0721B \tag{7.1}$$

Subsequently a block's size B is established to obtain the overlapping division of the image in blocks. Starting with the top-left corner, successive blocks are gathered by sliding left one pixel. Starting with the top-left corner, subsequent blocks are gathered by sliding left one pixel. Once all these blocks have been extracted, the process continues with one pixel down. The total number of blocks for an image of size $M \times N$ at the end of this process must be $(M - B + 1)(N - B + 1)$. Good results have been obtained with $B = 8$, which is the default value considered in what follows.

The next step consists of applying the DCT transform on every block to obtain a list of coefficients. DCT is chosen since many of the coefficients have values near 0, specifically, those corresponding to the highest frequencies, located near the bottom right corner of the coefficient matrix [FZWJ16]. This leads to the sorting of the coefficients following a zig-zag pattern as previously mentioned, shown in Figure 7.2.



1	2	6	7	15	16	28	
3	5	8	14	17	27		
4	9	13	18	26			
10	12	19	25				
11	20	24					
21	23						
22							

Figure 7.2: Zig-zag sorting of DCT coefficients

The zig-zag sorted lists are now truncated to k elements. The value of k is assigned according to the block size B (bigger blocks should consider more elements and vice versa). A truncation factor $0 < f_t < 1$ is fixed to compute k as shown in Equation 7.2.

$$k = \left[f_t B^2\right] \tag{7.2}$$

To accelerate the sorting process, only small integer values (k) will be considered. After truncation, the remaining values are quantized using a quantization factor f_q. Quantization is achieved by first dividing every value of the list by f_q and rounding the result. This process is specified in Equation 7.3. Values $a_{i1}, \ldots, a_{ik}$ denote the original coefficients of the i-th truncated list.

$$\vec{a_i} = \left(\left[\frac{a_{i1}}{f_q}\right], \left[\frac{a_{i2}}{f_q}\right], ..., \left[\frac{a_{ik}}{f_q}\right]\right) \tag{7.3}$$

With the truncated and quantized lists of coefficients, the next step is to sort them in lexicographic order. With this process, it is expected that very similar blocks provide similar quantized coefficient vectors.

After sorting, a matrix is produced with rows corresponding to similar blocks close together, but a similarity measure will indicate whether two blocks are duplicates of each other. For this measure, two thresholds S_t and T_t are previously defined ($S_t = 4$ and $T_t = 0.06$). The values for these thresholds are discussed later. The precise process used to decide if two blocks are the same is detailed in the following steps:

- Let $A = (a_{ij})$ be the sorted matrix of coefficients and $\vec{a_i}$ the i-th row of A. The first step is to define N as the maximum number of rows to be compared with $\vec{a_i}$. This is, $\vec{a_i}$ will be compared with $\vec{a}_{i+l}$ for $l = 1, \ldots, N$.
- Next, decide whether two blocks are identical using the pseudocode defined as Algorithm 3.
- After this process is completed, verify if $c < C_t$. In this case a_i and a_j are marked as a copy.

Algorithm 3: Pseudocode to decide when two blocks are identical

```
$r_{\max} \leftarrow -\infty$;
$r_{\min} \leftarrow \infty$;
$c \leftarrow 0$;
for $l = 1, \ldots, k$ do
    if $a_{lj} = 0$ & $|a_{li}| \geq S_t$ then
        $c \leftarrow c + 1$;
    else
        $r \leftarrow a_{il}/a_{jl}$
        if $r > r_{\max}$ then
            $r_{\max} \leftarrow r$
        if $r < r_{\min}$ then
            $r_{\min} \leftarrow r$
if $r_{\max} - r_{\min} \geq T_t$ then
    $c \rightarrow c + 1$
```

Since contiguous blocks are generally very similar, we should discard marking them as possible copies by selecting only blocks that are distant enough. For this, we consider the coordinates of the upper left pixel of a pair of similar blocks, namely, $v_i = (x_i, y_i)$ and $v_j = (x_j, y_j)$, corresponding to a_i, a_j respectively. If the Euclidean distance of v_i and v_j, computed following Equation 7.4, exceeds a given distance threshold (T_d), then the blocks are considered for the subsequent steps of the process, otherwise they are discarded.

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} > T_d \tag{7.4}$$

The final step of the detection algorithm consists of computing the frequencies for the transference vectors of every pair of suspicious blocks. The transference vector of two blocks starting at positions (x_i, y_i) and (x_j, y_j) respectively, is $\vec{v} = (x_i - x_j, y_i - y_j)$. If the frequency of a given transference vector is high, it is more likely that a big zone of the image has been displaced to the same region. If this frequency exceeds a given frequency threshold (T_f), in all the pairs of suspicious blocks associated with the transference vector, both blocks are marked. If the analysis is successful, a well-defined area must be perfectly visible, showing the original and copied regions. Unfortunately, the original region is not distinguished from the copied region.

7.2. Experimental Results

This section will show the experiments that have been performed to evaluate the effectiveness of the algorithm for identifying the duplicated region with copy-move techniques. Throughout the tests carried out, it has been possible to verify that the algorithm works with any format, such as JPEG, PNG, BMP, among others. It should also be noted that the image size does not influence the accuracy of the results; it only produces variations in the processing time, as shown in Section 7.2.5.

7.2.1. Datasets

The Image Manipulation Dataset [CRJ+12b] (hereinafter referred to as **D1**) is a ground truth database for benchmarking the detection of image tampering artifacts. It includes 48 base images, separate snippets from these images, and a software framework for creating ground truth data. The idea is to ”replay” copy-move forgeries by copying, scaling and rotating semantically meaningful image regions.

The CMFD GRIP Dataset by Cozzolino et al. [CPV15] (hereinafter referred to as **D2**) is a dataset composed of 80 images, with realistic copy-move forgeries. All these images have size 768x1024 pixels, while the forgeries have arbitrary shapes, aimed at obtaining visually satisfactory results.

The CoMoFoD database [TZGG13] (hereinafter referred as **D3** and **D4**) has 260 image sets, 200 images in the small image category (512x512), and 60 images in the large image category (3000x2000). In both categories, the following transformations are applied:

- Translation: a copied region is only translated to the new location without performing any transformation,
- Rotation: a copied region is rotated and translated to the new location,
- Scaling: a copied region is scaled and translated to the new location,
- Distortion: a copied region is distorted and translated to the new location

The distortion added to the dataset’s images can be Noise adding, Image blurring, Brightness change, Colour reduction, Contrast adjustments or a combination of two or more distortions on a copied region before moving it to the new location.

Ardizzone *et al.* [ABM15] make a copy-move forgery dataset (hereinafter referred to as **D5**) which contain medium-sized images, almost all 1000x700 or 700x1000. This dataset contains 50 uncompressed images with simply translated copies and 46 uncompressed images with 11 different types of rotation around the angle zero in the range of [-25°, 25°] with step 5° and 11 scaling factors in the range of [0.75, 1.25] with step 0.05.

The CMH dataset (hereinafter referred to as **D6**) was created by [SCFR15] and comprises 108 realistic cloning images. Each image is stored in the PNG format (which does not modify pixel values), and has a resolution varying from 845x634 pixels (the smallest) to 1296x972 pixels (the biggest). The dataset contains four groups of images:

- 23 images where the cloned area was only copied and then moved (simple case);
- 25 images with a rotation of the duplicated area (orientations in the range of 90° and 180°);
- 25 images with a resizing of the cloned area (scaling factors between 80% and 154%);
- 35 images involving rotation and resizing altogether.

7.2.2. Experiments Setup

In all the experiments carried out, *Python* has been used as a programming language, due to its great flexibility to perform data analysis. For the evaluation of the proposed algorithm, the experiments carried out in this chapter used the public datasets previously described and the Table 7.1 shows the main features of each dataset.

Table 7.1: Used dataset's features

Datasets	No. Images	Resolutions	Trnasformations	Formats
D1 [CRJ+12b]	48	2362x1581 3888x2592	None	jpg/png
D2 [CPV15]	80	1024x768	None	png
D3 [TZGG13]	960	3000x2000	Rotation, Scaling, Distortion	png/jpg
D4 [TZGG13]	200	512x512	Rotation, Scaling, Distortion	png/jpg
D5 [ABM15]	96	1024x768	Rotation, Scaling, Distortion	bmp
D6 [SCFR15]	108	845x634 1296x972	Rotation, Scaling, Distortion	png

The characteristics of the equipment in which the experiments were carried out are presented in Table 7.2. These are an essential factor to take into account since the execution times of the different tests vary according to the resources available.

Table 7.2: Features of the experimentation equipment

Resources	Features
Operating System	Ubuntu 17.04
Memory RAM	12 GB
Processor	Intel® Core™ i5-6200U CPU @ 2.30GHz x 4
Graphic Card	Intel® HD Graphics 520 (Skylake GT2)
Storage	64 GB

7.2.3. Truncation Factor Evaluation

This experiment assesses and verifies the effectiveness of the proposed algorithm. This algorithm makes use of different configurable parameters. Depending on the assigned value, the results may vary significantly. In [ZWWZ17] proposed a copy-move forgery detection algorithm based on DCT, which produced excellent results in the identification of copy-move manipulations. To perform their experiments, they made comparisons between the parameters used by other investigations. The values established by them are used in this research to initialize the parameters of the algorithm. Table 7.3 shows each of the parameters used and their corresponding values.

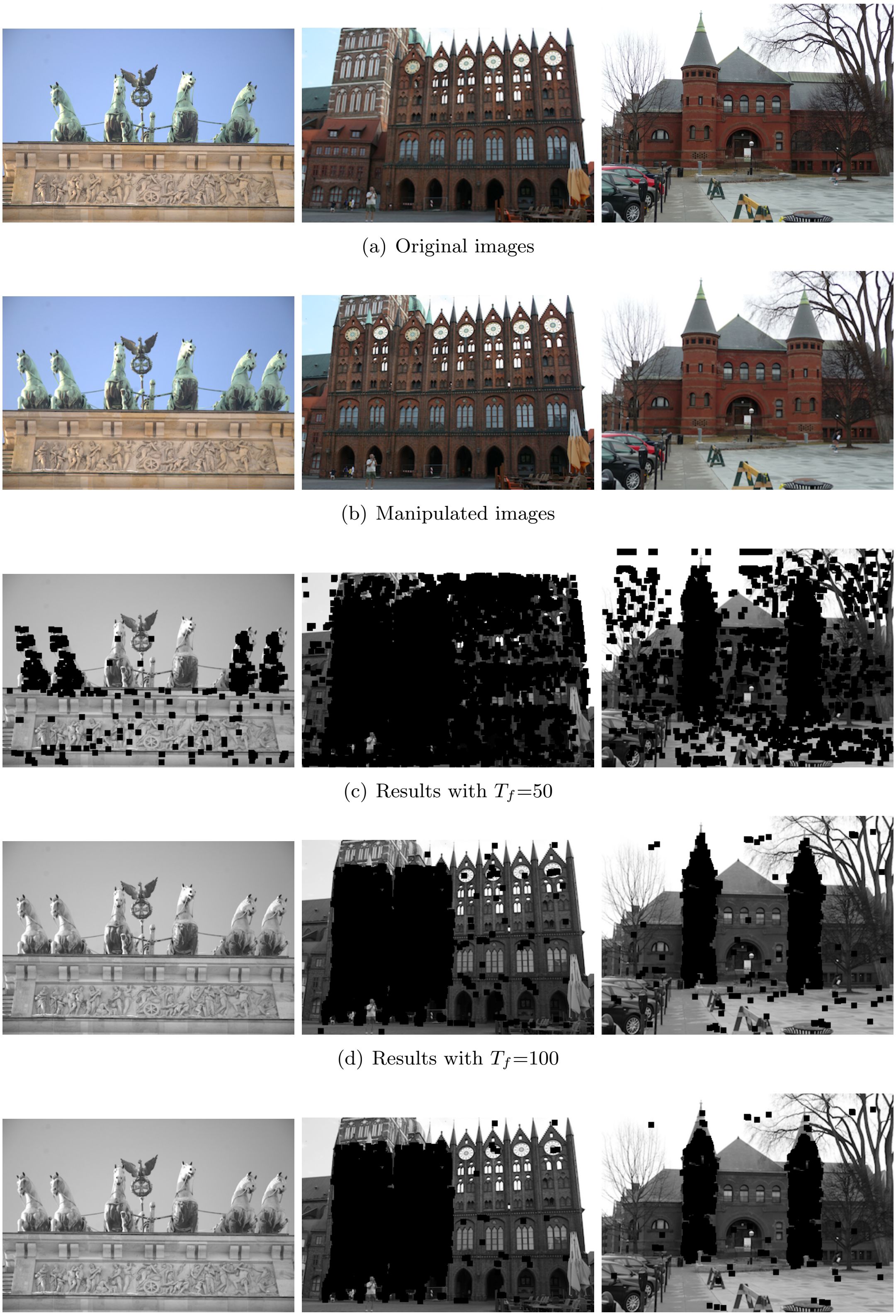
Table 7.3: Configurable parameters of the copy-move algorithm

Parameter	**Name**	**Assigned Value**
f_t	Truncation factor	0.25
f_q	Quantification factor	4
N_a	Comparable neighbouring rows	3
S_t	S Threshold	4
T_t	T Threshold	0.06
C_t	Similarity threshold	3
T_f	Frequency Threshold	50
T_d	Distance of vectors	20

The parameter that improves the results is the frequency threshold or T_f. This parameter sets the value under which a block of the image can be considered a valid manipulation. If a block appears several times in the image as a duplicate, and the frequency of appearance exceeds the one established by the threshold T_f, it will be considered as part of the manipulation. Because the image is segmented into overlapping blocks, it is possible to analyse the frequency of appearance of the duplicated blocks.

When T_f is high, the final results are more refined, removing the areas identified as manipulated that are false positives. In the experiment, the parameter T_f is set to three values: 50, 100, and 150. Figure 7.3 shows the results of the detection for these three T_f values.

From Figure 7.3, it is evident that at a higher value of T_f, the results present less noise (i.e. the black areas that are not part of the forgery). In the first image, the manipulation is identified at $T_f = 50$ (Figure 7.3(c)). With a higher value, the algorithm does not find any duplicate block that meets the frequency of appearance established by T_f. On the contrary, the other two images show that noise produced by false positives is removed when $T_f > 50$. This difference happens because, in both images, the forged areas occupy a significant proportion of the image so the frequency of appearance of the duplicated blocks would be, much higher than the overlap.



(a) Original images

(b) Manipulated images

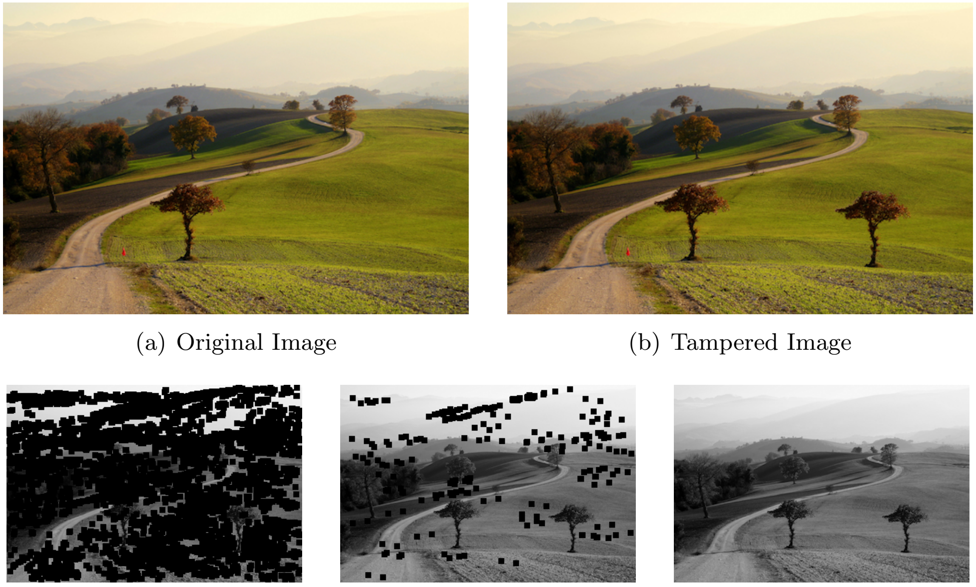
(c) Results with T_f=50

(d) Results with T_f=100

(e) Results with T_f=150

Figure 7.3: Detection of copy-move manipulations

However, the algorithm fails with a specific type of manipulation. For example, when parts of a duplicated block are modified from the initial block as shown in Figure 7.4. In Figure 7.4(a), the tree located in the central part has been duplicated. This tree has gaps between the branches which have been edited in the duplication so that it integrates perfectly into the background. That is why the algorithm treats both trees as different objects and is not able to detect the forgery.



(a) Original Image

(b) Tampered Image

(c) Results with T_f=50

(d) Results with T_f=100

(e) Results with T_f=150

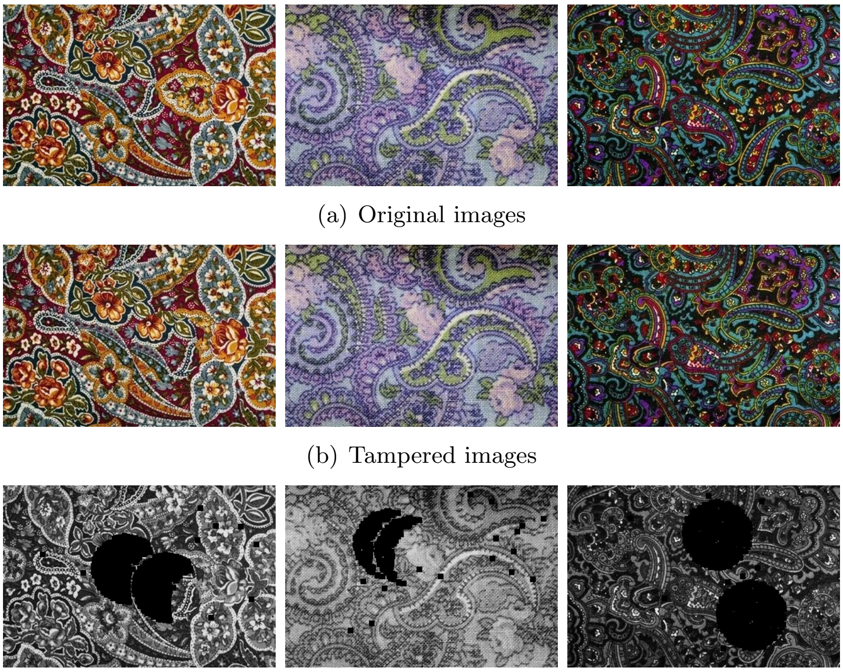
Figure 7.4: Duplicate area with details of the original image

7.2.4. Texture Influence on the Success Rate

In the second experiment, we checked the accuracy of the algorithm to detect the copy region on textures with similar patterns. In case, the manipulation goes unnoticed due to its excellent integration with the original background as is the case of images with the same colour pattern. Two tests were carried out with this type of image, and the parameter T_f was adjusted to the value 150 to reduce the noise of black dots in the results.

In the first test, images with multiple colours and details but similar patterns were used; this makes the duplicated area difficult to detect. Three examples of identification in this type of images are shown in Figure 7.5. As noted, the algorithm achieves remarkable accuracy.

For the second test, we used images where the duplicated region was moved to an area with the same colour as other regions of the image. In this type of images, it is also difficult to detect the duplicated region since it can be confused with another original region that has the same colour. Figure 7.6 shows three examples where the algorithm has an excellent performance in this type of manipulation.

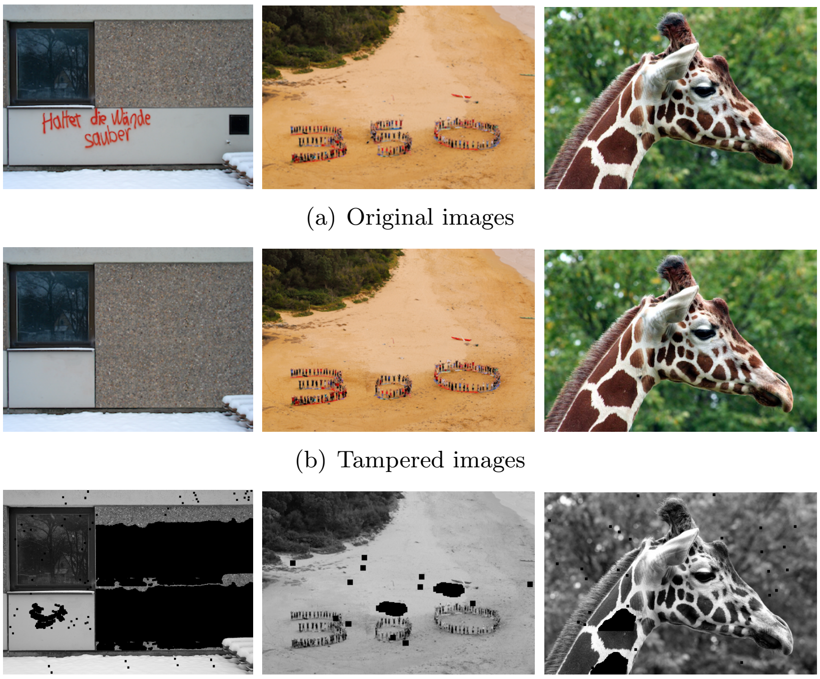


(a) Original images

(b) Tampered images

(c) Detection results

Figure 7.5: Copy-move detection over images with similar patterns textures



(a) Original images

(b) Tampered images

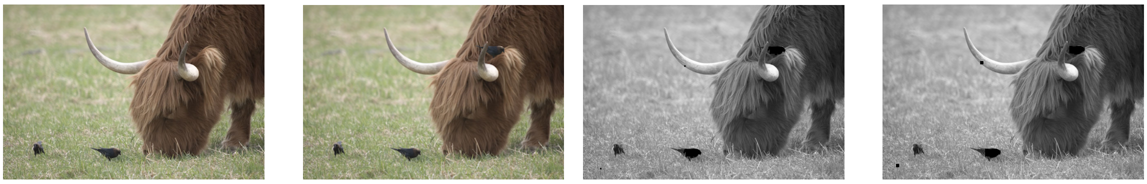
(c) Detection results

Figure 7.6: Copy-move detection over images with areas of the same color

7.2.5. Image Resolution Influence

In this experiment, we analysed the efficiency of the algorithm in large and high-resolution images. We observed that when scaling an image to a smaller size, the accuracy of the algorithm remains high without undergoing significant changes. Thus, it is possible to scale large images before the algorithm processes them, which increases efficiency without losing quality in the results.

Figure 7.7 shows an image modified by the copy-move forgery. In this image, the bird above the grass has been copied and placed onto the cow's head. The original image size is 1080x854 pixels (c), and the size of the scaled image is 640x427 pixels. The execution time it took for the algorithm to process the original image was 160 seconds, while the scaled image took 48 seconds. From Figure 7.7, it is apparent that the manipulation has been correctly detected in both images, so it is possible to perform the scaling without affecting the accuracy of the algorithm and considerably improving the execution time.



(a) Original Image (b) Tampered Image (c) Image of 1280 × 854 (d) Image of 640 × 427

Figure 7.7: Copy-move identification in scaled images

7.2.6. Comparison with Other Techniques on the State-of-the-Art

In order to test the proposed algorithm we use the six previously described datasets, and the metrics used to quantify its accuracy were the Precision, Recall, and F1 scores. The precision, P, is the ratio of the probability that a detected region is accurate, and the Equation 7.5 calculates the precision.

$$P = \frac{TP}{TP + FP} \tag{7.5}$$

where True Positives (TP) is the number of true positives pixels and False Positives (FP) the number of false positive pixels detected by the algorithm.

The Recall is the True Positive Rate (TPR) component. These are given by the recall, R, the true positive ratio which measure the ability of the algorithm to find all the positive samples. Its formula is the Equation 7.6.

$$R = \frac{TP}{TP + FN} \tag{7.6}$$

where TP is the number of true positives and False Negatives (FN) the number of false negatives

The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score is equal. F1 score is computed with Equation 7.7.

$$F1 = \frac{2PR}{P + R} \tag{7.7}$$

where P is the precision and R the recall obtained by the algorithm on each analysed image.

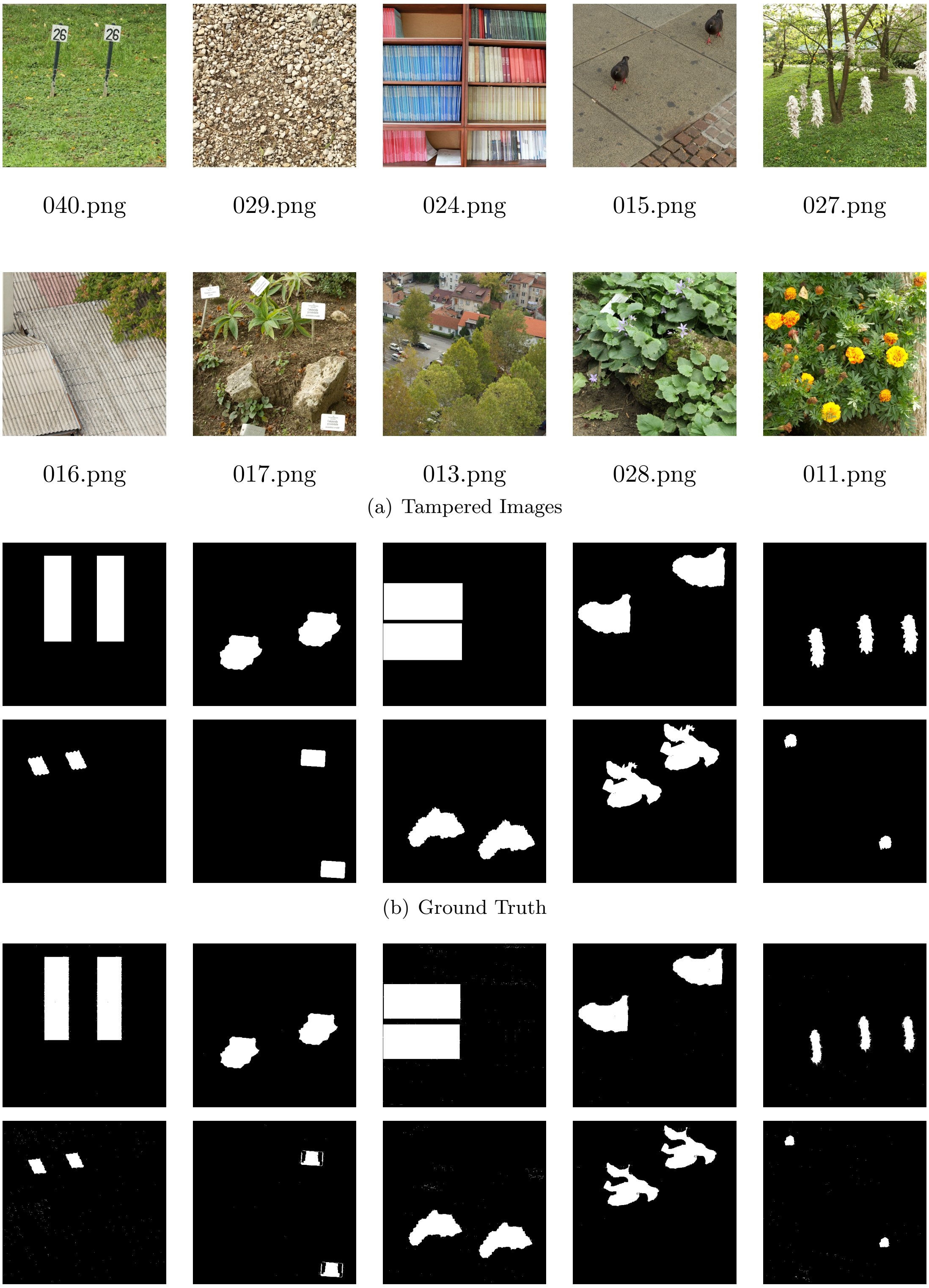
The results summarised in Table 7.4 shows a high accuracy precision and recall over all the tested datasets, especially datasets **D5**. Our proposed algorithm gets a precision average of over 93% even on images that contain distortion, such as an increase or decrease in brightness and/or contrast, and small geometrical transformations like slight degree rotation. To validate our results, it is essential to compare our algorithm to a related method, such as the one proposed by Alkawaz *et al.* in [ASSR18] in which the authors get a 96% of recall and a 64,52% of precision using a block size of 8x8. Figure 7.8 and Table 7.5 show the outputs and the results and the outputs.

Table 7.4: Results obtained by proposed algorithm

Datasets	Precision	Recall	F1
D1 [CRJ$^+$12b]	95.51%	69.74%	78.7%
D2 [CPV15]	90.61%	58.71%	70.31%
D3 [TZGG13]	95.35%	70.99%	79.51%
D4 [TZGG13]	86.66%	57.25%	67.06%
D5 [ABM15]	96.58%	**81.06%**	**87.94%**
D6 [SCFR15]	**97.52%**	27.49%	42.52%
Average	93.71%	60.87%	71.01%

Table 7.5: Results obtained by proposed algorithm

Images	**Our algorithm**		**[ASSR18]**	
D4 [TZGG13]	**Precision**	**Recall**	**Precision**	**Recall**
040.png	99.93%	86.95%	**100%**	97.53%
029.png	**99.96%**	80.58%	95.19%	96.25%
024.png	99.59%	**91.31%**	49.80%	**100%**
015.png	99.41%	85.22%	87.62%	99.48%
027.png	98.71%	63.07%	63.32%	88.47%
016.png	91.78%	67.07%	62.53%	97.3%
017.png	99.35%	55.15%	59.51%	99.86%
013.png	99.01%	81.19%	59.25%	98.68%
028.png	99.86%	71.79%	41.49%	93.37%
011.png	91.29%	58.11%	26.58%	94.85%
Average	97.88%	74.04%	64.53%	96.58%



(a) Tampered Images

(b) Ground Truth

(c) Detection Results

Figure 7.8: Copy-move detection over images with areas of the same color

7.3. Summary

As the the famous saying goes, “A picture is worth a thousand words”. Therefore, faster and reliable algorithms to analyse the integrity of an image are needed. Nowadays, thanks to the fast and easy way of sharing images, plus the ease of use in professional image-editing tools, it is harder to detect forgeries

In this section, a new approach for forgery detection was presented. The experiments carried out with the proposed algorithm have shown their robustness and efficiency in the results obtained. The algorithm can detect and locate, with high precision, the duplicated zone in the image. Besides its accuracy, the algorithm proved to be a fast method for analysing even high-resolution photos in a short time, unlike what can be found in the state-of-the-art techniques.

Chapter 8

Conclusions and Future Work

The aim of this work has been to make progress in the field of forensic analysis of digital images, placing special emphasis on the so-called passive techniques that aim to corroborate the authenticity of digital images, detecting any manipulation of their content. More specifically, the techniques developed here are intended to detect manipulations of the Copy and Move type and the Splice type, as they are the most common. The starting point for this thesis was an exhaustive review of the current most up to date type of technique, analyzing whether some of them could be improved, as it was also intended that at the end of the thesis there would be a fully operational forensic tool for police use. The contributions of this thesis are summarized in 5 forensic techniques described below:

The first of these is a technique designed to be the starting point for forensic analysis of digital images. The proposed technique is based on the combination of the wavelet transform with histograms extracted from local binary patterns for manipulation detection. This technique uses a vectorial support machine as a classification method. For a closer look, the technique consists of extracting the wavelet characteristics from the image that, later, are analyzed with the local binary pattern and, finally, the histogram is calculated to the resulting blocks. Then, and in order to improve the accuracy, the QMF filter is applied, obtaining the characteristics that serve to train the vectorial support machine to later classify a set of images between original and manipulated.

To evaluate this technique, several experiments were carried using a set of images which are widely used in the most up to date research. The results showed that the technique obtained a maximum accuracy of 99.43% on the CASIA v2.0 database, improving on the existing literature. Furthermore, it proved to be efficient in the tests carried out after comparing its execution time with those existing in current research. This first contribution has made it possible to achieve the second objective set out at the beginning of this thesis: to develop a technique that can automatically classify images as manipulated or original.

Thanks to its integration with the RAMSES platform, images can come from a variety of sources, from images contained in traditional storage devices such as hard disks and external memories to images from social networks. However, within a process of forensic analysis of digital images, it is not enough to determine if it is original or if there are indications of manipulation. It is also necessary to be able to identify the type of manipulation and the manipulated region of the image. This is the context of the remaining four contributions of this thesis, which achieve the objectives of identifying splice-type and copy-and-move manipulations and locating the manipulated region within the image.

Three forensic techniques have been developed to identify and locate splice-type manipulations.

Two of these three techniques are based on the analysis of the traces left by manipulations on images with JPEG compression. More specifically, one of them is based on the characteristics left by JPEG loss compression using the error level analysis technique which is able to highlight the pixels that contain a different level of compression from the rest of the image. Based on this, the areas that do not belong to the original content are marked. The second detects manipulations in a colour image using chromatic interpolation algorithms in the following way: it divides the image into overlapping blocks and estimates the interpolation pattern and the average square error of these blocks, analyzing the anomalies found and identifying whether or not there is manipulation in the image at the pixel level, thus allowing the manipulated region to be located.

These two techniques were evaluated with public data sets and those widely used in up to date research. The experiments carried out for the first technique used the CASIA v1.0 database as well as its own data set, showing that the image quality has a direct influence on the results, because, in the case of CASIA, which has a low-resolution quality, the technique found some difficulties to detect the region that does not belong to the original image. However, in the case of its own data set containing high-resolution images, the technique achieved an accuracy of 73.3%. Furthermore, thanks to the speed of this technique, it can be very useful for the forensic analyst when carrying out analysis of large amounts of images and can be the first filter to continue analyzing images detected as manipulated with more sophisticated and robust, but rather slower techniques.

The experiments carried out for the second technique were successful in introducing images with dimensions greater than 1500x1500 pixels. However, images, where there are scenes with very large white regions, generate false positives because the variance calculated in those sections is very low with respect to the rest of the image. On the other hand, when analyzing images with resolutions less than or equal to 700x700 pixels, the technique has difficulty in detecting the area with modifications, since the image information is not sufficient to make the difference between the known variances. Despite this, this technique, as in the previous case, can serve as a first filter to separate from a large number of images, those most susceptible to having been manipulated and, subsequently, use another more precise technique that performs a more exhaustive analysis The third technique identifies manipulations, both Splice and Copy & Move because it uses the marks left by the camera's colour filter matrix at the time of generating the captured image. The technique allows the detection of manipulations to different colour filters to be generalized without losing precision, as is the case with similar techniques that work in the RGB colour space. This technique can be applied in different scenarios as long as marks are periodically present in the analyzed image. Due to its configuration, it uses different bands and a smaller block size than those proposed in the literature, allowing it to improve the precision when identifying the manipulated region. In addition, by using colour bands which are different from green, it allows for the identification of manipulations even if some filter has been applied to the image after it has been manipulated. The experiments performed to validate this technique used the RAISE dataset, and the CMFD dataset.

The fourth and final technique identifies Copy and Move type manipulations and, after performing a block analysis of the image, allows it to locate the manipulated region with an average accuracy of 97.88%. The data set used consists of a total of 1492 images with different resolutions and post-processing. The analyzed images containing post-processing were those that gave the highest amount of false positives and false negatives. However, the technique allowed the identification of manipulations after the application of colour filters and slight geometric transformations (rotations, size changes, etc.).

8.1. Future Works

Forensic analysis of digital images, and especially the detection of image falsification, is undoubtedly a broad and current research discipline. As image editing applications become increasingly sophisticated and user-friendly, it is easier to create more realistic and convincing counterfeits. Also, as counterfeit detection techniques become more widely known, vulnerabilities will be discovered and exploited to strengthen counterfeiting techniques. For all the above reasons, and although the passive forensic techniques for detecting digital image manipulations proposed in this thesis have achieved satisfactory results, much remains to be done. Some potential future lines of investigation are summarized below:

- **Expanding the types of manipulations detected**: The techniques developed in this thesis are based on the analysis of compression with loss (Chapter 5) and on the analysis of pixels (Chapters 4, 6, 7 and 8), with the aim of detecting manipulations of the Copy-Paste and Splice types. These techniques can be combined with any of the other passive techniques presented in Chapter 3 for Resampling and Source Tampering. This type of combination could significantly increase the results obtained thanks to the diversity of the techniques used.

- **Extend Copy-Move-Splice tampering detection techniques to digital video**: As interest in passive forensic image analysis grows, the scientific community is also interested in applying and extending these techniques to the problem of fake video. Frame by frame analysis of a video thus becomes the immediate application; however, the temporary nature of the videos adds additional complexity and opens up new avenues of study.

- **Apply deep learning techniques to automatic manipulation detection**: Recently, deep learning techniques have proven to be effective and with remarkable performance in artificial intelligence and computer vision applications. It would be interesting to run and compare the automatic manipulation detection technique in Chapter 4, which uses supervised learning techniques with deep-learning techniques and also increases the size of the training data sets used.
- **Improve the effectiveness and robustness of techniques based on the analysis of JPEG compression with the use of automatic denoising filters**: The main limitations of the error level analysis technique, developed in Chapter 5, are its limited application to images subjected to lost compression and that the final decision is made by the forensic analyst when interpreting the results of the detection. Therefore, including an adaptive filter to remove the noise generated by the compression scheme used could improve the error level. Successfully attenuating this noise would improve error levels by better identifying regions of the image where manipulation has occurred. This would facilitate the task of the forensic analyst.
- **Explore additional features present in the images to identify manipulations in more detail**: While the Copy and Move manipulation detection techniques in Chapters 6 and 7 provide very good results, the use of local invariant characteristic and descriptor extraction techniques can be studied to provide a more accurate location of manipulated regions in low spatial frequency images (e.g. smooth regions with minimal texture).

Bibliography

[ABC$^+$11] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, and G. Serra. A SIFT-Based Forensic Method for Copy-Move Attack Detection and Transformation Recovery. *IEEE Transactions on Information Forensics and Security*, 6(3):1099–1110, September 2011.

[ABM15] E. Ardizzone, A. Bruno, and G. Mazzola. Copy–move Forgery Detection by Matching Triangles of Keypoints. *IEEE Transactions on Information Forensics and Security*, 10(10):2084–2094, October 2015.

[AELTZ20] E.I. Abd El-Latif, A. Taha, and H.H. Zayed. A Passive Approach for Detecting Image Splicing Based on Deep Learning and Wavelet Transform. *Arabian Journal for Science and Engineering*, 31(11):3379—-3386, February 2020.

[AH17] A. Alahmadi and M. Hussain. Passive Detection of Image Forgery Using DCT and Local Binary Pattern. *Signal, Image and Video Processing*, 11(1):81–88, January 2017.

[AHA$^+$13] A. A. Alahmadi, M. Hussain, H. Aboalsamh, G. Muhammad, and G. Bebis. Splicing Image Forgery Detection Based on DCT and Local Binary Pattern. In *Proceedings of the IEEE Global Conference on Signal and Information Processing*, pages 253–256, Austin, USA, December 2013.

[ASSR18] M. H. Alkawaz, G. Sulong, T. Saba, and A. Rehman. Detection of Copy-Move Image Forgery Based on Discrete Cosine Transform. *Neural Computing and Applications*, 30(1):183–192, July 2018.

[AVGFSOGV20a] E. A. Armas Vega, E. González Fernández, A. L. Sandoval Orozco, and L. J. García Villalba. Image Tampering Detection by Estimating Interpolation Patterns. *Future Generation Computer Systems*, 107:229–2327, June 2020.

[AVGFSOGV20b] E. A. Armas Vega, E. González Fernández, A. L. Sandoval Orozco, and L. J. García Villalba. Passive Image Forgery Detection Based on the Demosaicing Algorithm and JPEG Compression. *IEEE Access*, 8:11815–11823, January 2020.

[AVMHPA$^+$19] E. A. Armas Vega, L. A. Martínez Hernández, S. Pérez Arteaga, A. L. Sandoval Orozco, and L. J. García Villalba. Algoritmo de Interpolación Cromática para la Detección de Zonas Manipuladas de Imágenes. In *Actas de las V Jornadas Nacionales de Ciberseguridad*, Cáceres, Spain, June 2019.

[AVMHPA$^+$20] E. A. Armas Vega, L. A. Martínez Hernández, S. Pérez Arteaga, A. L. Sandoval Orozco, and L. J. García Villalba. Detección de Manipulaciones

Copy-Move en Ficheros Multimedia mediante la Transformada Discreta del Coseno. In *Actas de el X Congreso Iberoamericano de Seguridad Informática*, Bogotá, Colombia, January 2020.

[AVQHSOGV19] E. A. Armas Vega, C. Quinto Huamán, A. L. Sandoval Orozco, and L. J. García Villalba. Técnica de Autenticación de Imágenes Digitales Basada en la Extracción de Características. In *Actas de las V Jornadas Nacionales de Ciberseguridad*, Cáceres, Spain, June 2019.

[AVSOVHC18] E. A. Armas Vega, A. L. Sandoval Orozco, García L. J. Villalba, and J. Hernández-Castro. Digital Images Authentication Technique Based on DWT, DCT and Local Binary Patterns. *Sensors*, 18(10):3372, October 2018.

[BJGY10] X. Bo, W. Junwen, L. Guangjie, and D. Yuewei. Image Copy-Move Forgery Detection Based on SURF. In *Proceedings of the International Conference on Multimedia Information Networking and Security*, pages 889–892, Nanjing, China, November 2010.

[BM13] G. K. Birajdar and V. H. Mankar. Digital Image Forgery Detection Using Passive Techniques: A Survey. *Digital Investigation*, 10(3):226–245, October 2013.

[BM16] G. K. Birajdar and V. H. Mankar. Passive Method for Rescale Detection Using Quadrature Mirror Filter Based Higher Order Statistical Features. *International Journal of Wavelets, Multiresolution and Information Processing*, 14(05):1650033–1–1650033–28, July 2016.

[BSVB16] A. Bharati, R. Singh, M. Vatsa, and K. W. Bowyer. Detecting Facial Retouching Using Supervised Deep Learning. *IEEE Transactions on Information Forensics and Security*, 11(9):1903–1913, September 2016.

[CDR13] C. Chen, A. Dantcheva, and A. Ross. Automatic Facial Makeup Detection with Application in Face Recognition. In *Proceedings of the International Conference on Biometrics (ICB)*, pages 1–8, Madrid, Spain, June 2013.

[CFGL08] M. Chen, J. Fridrich, M. Goljan, and J. Lukas. Determining Image Origin and Integrity Using Sensor Noise. *IEEE Transactions on Information Forensics and Security*, 3(1):74–90, March 2008.

[Cha04] A. Chambolle. An Algorithm for Total Variation Minimization and Applications. *Journal of Mathematical Imaging and Vision volume*, 20(1–2):89–97, January 2004.

[CL11] C. C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27–1–27–27, May 2011.

[CMB$^+$07] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker. *Digital Watermarking and Steganography.* Morgan kaufmann, November 2007.

[Con11] V. Conotter. *Active and Passive Multimedia Forensics.* PhD thesis, University of Trento, 2011.

[CPF04] A. C Popescu and H. Farid. Exposing Digital Forgeries by Detecting Duplicated Image Regions. *Department of Computer Science*, 646:1–11, January 2004.

[CPV15] D. Cozzolino, G. Poggi, and L. Verdoliva. Efficient Dense-Field Copy–Move Forgery Detection. *IEEE Transactions on Information Forensics and Security*, 10(11):2284–2297, November 2015.

[CR19] B. Chaitra and P. V. B. Reddy. A Study on Digital Image Forgery Techniques and its Detection. In *Proceedings of the International Conference on contemporary Computing and Informatics (IC3I)*, pages 127–130. IEEE, 2019.

[CRJ+12a] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou. An Evaluation of Popular Copy–Move Forgery Detection Approaches. *IEEE Transactions on Information Forensics and Security*, 7(6):1841–1854, December 2012.

[CRJ+12b] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou. An Evaluation of Popular Copy-Move Forgery Detection Approaches. *IEEE Transactions on Information Forensics and Security*, 7(6):1841–1854, December 2012.

[CSE00] C. Christopoulos, A. Skodras, and T. Ebrahimi. The JPEG2000 Still Image Coding System: An Overview. *IEEE Transactions on Consumer Electronics*, 46(4):1103–1127, November 2000.

[DCR12] A. Dantcheva, C. Chen, and A. Ross. Can Facial Cosmetics Affect the Matching Accuracy of Face Recognition Systems? In *Proceedings of the IEEE 5th International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 391–398, Washington DC, USA, September 2012. IEEE.

[DD00] W. K. L. Dickson and A. Dickson. *History of the Kinetograph, Kinetoscope, and Kineto-Phonograph.* The Museum of Modern Art, July 2000.

[DM09] A. E. Dirik and N. Memon. Image Tamper Detection Based on Demosaicing Artifacts. In *Proceedings of the 16th IEEE International Conference on Image Processing*, pages 1497–1500, Cairo, Egypt, November 2009.

[dSM05] E. A. B. da Silva and G. V. Mendonça. Digital Image Processing. In W. K. Chen, editor, *The Electrical Engineering Handbook*, pages 891 – 910. Academic Press, Burlington, 2005.

[DWT13] J. Dong, W. Wang, and T. Tan. CASIA Image Tampering Detection Evaluation Database. In *Proceedings of the IEEE China Summit and International Conference on Signal and Information Processing*, pages 422–426, Beijing, China, July 2013.

[DZDS18] F. Ding, G. Zhu, W. Dong, and Y. Q. Shi. An Efficient Weak Sharpening Detection Method for Image Forensics. *Journal of Visual Communication and Image Representation*, 50:93–99, January 2018.

[DZY+15] F. Ding, G. Zhu, J. Yang, J. Xie, and Y. Shi. Edge Perpendicular Binary Coding for USM Sharpening Detection. *IEEE Signal Processing Letters*, 22(3):327–331, March 2015.

[FBDRP12] P. Ferrara, T. Bianchi, A. De Rosa, and A. Piva. Image Forgery Localization via Fine-Grained Analysis of CFA Artifacts. *IEEE Transactions on Information Forensics and Security*, 7(5):1566–1577, October 2012.

[FCD12] X. Feng, I. J Cox, and G. Doerr. Normalized Energy Density-Based Forensic Detection of Resampled Images. *IEEE Transactions on Multimedia*, 14(3):536–545, June 2012.

[FLJ$^+$20] B. Feng, X. Li, Y. Jie, C. Guo, and H. Fu. A novel Semi-Fragile Digital Watermarking Scheme for Scrambled Image Authentication and Restoration. *Mobile Networks and Applications*, 25(1):82–94, January 2020.

[FSL03] J. Fridrich, D. Soukal, and J. Lukas. Detection of Copy Move Forgery in Digital Images. In *Proceedings of the Digital Forensic Research Workshop*, pages 5–8, Binghamton, New York, August 2003.

[Fuj] Fujifilm Global. FUJIFILM X20 - Feature: Focused on Bringing you the Ultimate Image Quality. 2/3-inch 12M X-Trans CMOS II Sensor — X Series — Digital Cameras — Fujifilm USA. https://www.fujifilmusa.com/products/digital_cameras/x/fujifilm_x20/features/page_02.html, Last accessed on 2019-12-17.

[FZWJ16] Q. Fu, X. Zhou, C. Wang, and B. Jiang. Mathematical Relation between APBT-Based and DCT-Based JPEG Image Compression Schemes. *Journal of Communications*, 11(1):84–92, January 2016.

[GFSOGVHC18] E. González Fernández, A. L. Sandoval Orozco, L. J. García Villalba, and J. Hernández Castro. Digital Image Tamper Detection Technique Based on Spectrum Analysis of CFA Artifacts. *Sensors*, 18(9):2804, August 2018.

[GVSORCHC17] L. J. García Villalba, A. L. Sandoval Orozco, J. Rosales Corripio, and J. Hernández Castro. A PRNU-Based Counter-Forensic Method to Manipulate Smartphone Image Source Identification Techniques. *Future Generation Computer Systems*, 76:418–427, November 2017.

[HGZ08] H. Huang, W. Guo, and Y. Zhang. Detection of Copy-Move Forgery in Digital Images Using SIFT Algorithm. In *Proceedings of the IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, volume 2, pages 272–276, December 2008.

[HHS10] F. Huang, J. Huang, and Y. Q. Shi. Detecting Double JPEG Compression with the Same Quantization Matrix. *IEEE Transactions on Information Forensics and Security*, 5(4):848–856, December 2010.

[HPME16] J. G. Han, T. H. Park, Y. H. Moon, and I. K. Eom. Efficient Markov Feature Extraction Method for Image Splicing Detection Using Maximization and Threshold Expansion. *Journal of Electronic Imaging*, 25(2):023031–1–023031–9, April 2016.

[HSA$^+$14] M. Hussain, S. Q. Saleh, H. Aboalsamh, G. Muhammad, and G. Bebis. Comparison between WLD and LBP Descriptors for Non-Intrusive Image Forgery Detection. In *Proceedings of the IEEE International Symposium on*

Innovations in Intelligent Systems and Applications (INISTA), pages 197–204, Alberobello, Italy, June 2014.

[KAD15] N. Kose, L. Apvrille, and J. L. Dugelay. Facial Makeup Detection Technique Based on Texture and Shape Analysis. In *Proceedings of the 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 1, pages 1–7, Ljubljana, Slovenia, May 2015.

[KF11] E. Kee and H. Farid. A Perceptual Metric for Photo Retouching. *National Academy of Sciences*, 108(50):19907–19912, December 2011.

[KG16] M. Kaur and S. Gupta. A Passive Blind Approach for Image Splicing Detection based on DWT and LBP Histograms. In *Proceedings of Sixth International Symposium on Security in Computing and Communication*, pages 318–327, Bangalore, India, September 2016. Springer.

[KH17] P. Korus and J. Huang. Multi-Scale Analysis Strategies in PRNU-Based Tampering Localization. *IEEE Transactions on Information Forensics and Security*, 12(4):809–824, April 2017.

[KKI17] N. Khetan, L. Kejriwal, and S. Indu. Enhancement of Degraded Manuscript Images using Adaptive Gaussian Thresholding. *International Journal of Future Generation Communication and Networking*, 10(1):47–60, October 2017.

[KMC$^+$07] N. Khanna, A. K. Mikkilineni, G. T. Chiu, J. P. Allebach, and E. J. Delp. Forensic Classification of Imaging Sensor Types. In *Proceedings of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX*, volume 6505, pages 65050U–65050U–9, San Jose, California, USA, February 2007.

[KW08] X. Kang and S. Wei. Identifying Tampered Regions Using Singular Value Decomposition in Digital Image Forensics. In *Proceedings of the International Conference on Computer Science and Software Engineering*, volume 3, pages 926–930, December 2008.

[LCWH17] P. Liu, K. K. R. Choo, L. Wang, and F. Huang. SVM or Deep Learning? A Comparative Study on Remote Sensing Image Classification. *Soft Computing*, 21(23):7053–7065, July 2017.

[Lis07] Listverse. Top 15 Photoshopped Photos That Fooled Us All. http://listverse.com/2007/10/19/top-15-manipulated-photographs/, October 2007.

[M.09] Pierre M. *Multiscale Image Decompositions and Wavelets*, pages 123–142. Academic Press, Boston, 2009.

[Mal77] D. F. Malin. Unsharp Masking. *American Astronomical Society Photo Bulletin*, 16:10–13, 1977.

[Mar90] J.L. Marignier. Historical Light on Photography. *Nature*, 346(6280):115–115, July 1990.

[MC11] D. Menon and G. Calvagno. Color Image Demosaicking: An Overview. *Signal Processing: Image Communication*, 26(8):518–533, October 2011.

[MHB12] G. Muhammad, M. Hussain, and G. Bebis. Passive Copy Move Image Forgery Detection using Undecimated Dyadic Wavelet Transform. *Digital Investigation*, 9(1):49–57, June 2012.

[MJN19] Z. Moghaddasi, H. A Jalab, and R. M. Noor. Image Splicing Forgery Detection based on Low-Dimensional Singular Value Decomposition of Discrete Cosine Transform Coefficients. *Neural Computing and Applications*, 31(11):7867–7877, July 2019.

[MS10] B. Mahdian and S Saic. A Bibliography on Blind Methods for Identifying Image Forgery. *Signal Processing: Image Communication*, 25(6):389–399, July 2010.

[NC04] T. T. Ng and S. F. Chang. A Data Set of Authentic and Spliced Image Blocks. Technical report, Columbia University, June 2004.

[NKK15] N. Nirmalkar, S. Kamble, and S. Kakde. A Review of Image Forgery Techniques and their Detection. In *Proceedings of the International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, pages 1–5. IEEE, 2015.

[NL19] E. Najafi and K. Loukhaoukha. Hybrid Secure and Robust Image Watermarking Scheme based on SVD and Sharp Frequency Localized Contourlet Transform. *Journal of information security and applications*, 44:144–156, February 2019.

[NLZN19] Y. Niu, X. Li, Y. Zhao, and R. Ni. An Enhanced Approach for Detecting Double JPEG Compression with the Same Quantization Matrix. *Signal Processing: Image Communication*, 76:89 – 96, August 2019.

[NYC15] A. Nguyen, J. Yosinski, and J. Clune. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 427–436, 2015.

[OPM02] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, July 2002.

[Ots79] N. Otsu. A Threshold Selection Method from Gray–Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, January 1979.

[PC18] C. S. Park and J. Y. Choeh. Fast and Robust Copy-Move Forgery Detection based on Scale-Space Representation. *Multimedia Tools and Applications*, 77(13):16795–16811, July 2018.

[PF05] A. C. Popescu and H. Farid. Exposing Digital Forgeries in Color Filter Array Interpolated Images. *IEEE Transactions on Signal Processing*, 53(10):3948–3959, October 2005.

[RPH14] A. Rocha, A. Piva, and J. Huang. The First Image Forensics Challenge. http://ifc.recod.ic.unicamp.br/, January 2014.

[SA17] V Santhi and DP Acharjya. Improving the Security of Digital Images in Hadamard Transform Domain Using Digital Watermarking. In *Biometrics: Concepts, Methodologies, Tools, and Applications*, pages 1592–1618. IGI Global, 2017.

[SCC07] Y. Q. Shi, C. Chen, and W. Chen. A Natural Image Model Approach to Splicing Detection. In *Proceedings of the 9th workshop on Multimedia & security*, pages 51–62, Dallas, USA, September 2007.

[SCFR15] E. Silva, T. Carvalho, A. Ferreira, and A. Rocha. Going Deeper Into Copy-Move Forgery Detection: Exploring Image Telltales Via Multi-Scale Analysis and Voting Processes. *Journal of Visual Communication and Image Representation*, 29:16–32, May 2015.

[SDJ+18] S. Sadeghi, S. Dadkhah, H. A. Jalab, G. Mazzola, and D. Uliyan. State of the art in Passive Digital Image Forgery Detection: Copy-Move Image Forgery. *Pattern Analysis and Applications*, 21(2):291–306, December 2018.

[SE18] A. Shah and E. El-Alfy. Image Splicing Forgery Detection Using DCT Coefficients with Multi-Scale LBP. In *Proceedings of the International Conference on Computing Sciences and Engineering (ICCSE)*, pages 1–6, Kuwait City, Kuwait, March 2018.

[SHD14] R. Sarikaya, G. E. Hinton, and A. Deoras. Application of Deep Belief Networks for Natural Language Understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(4):778–784, February 2014.

[SOGVAG+15] A. L. Sandoval Orozco, L. J. García Villalba, D. M. Arenas González, J. Rosales Corripio, J. C. Hernández Castro, and S. Gibson. Smartphone Image Acquisition Forensics Using Sensor Fingerprint. *IET Computer Vision*, 9(5):723–731, October 2015.

[SVOZMR18] J. M. Sánchez-Vigil, M. Olivera-Zaldua, and J. C. Marcos-Recio. Patentes sobre Fotografía en España (1839-1939). Análisis Documental: Contenidos y Solicitantes. *Revista Española de Documentación Científica*, 41(3):210, September 2018.

[Tac12] M. Tachi. Image Processing Device, Image Processing Method, and Program Pertaining to Image Correction, November 2012.

[TM08] S. Tatiraju and A. Mehta. Image Segmentation using K-Means Clustering, EM and Normalized Cuts. *Department of EECS*, 1:1–7, 2008.

[TU19] S. Teerakanok and T. Uehara. Copy-Move Forgery Detection: A State-of-the-Art Technical Review and Analysis. *IEEE Access*, 7:40550–40568, March 2019.

[TZGG13] D. Tralic, I. Zupancic, S. Grgic, and M. Grgic. CoMoFoD—New Database for Copy-Move Forgery Detection. In *Proceedings of the 55th International Symposium in Electronics in Marine*, pages 49–54, Zadar, Croatia, September 2013.

[WC04] C. Y. Wen and C. M. Chou. Color Image Models and its Applications to Document Examination. *Forensic Science Journal*, 3(1):23–32, January 2004.

[WDT10] W. Wang, J. Dong, and T. Tan. Image Tampering Detection Based on Stationary Distribution of Markov Chain. In *Proceedings of the IEEE International Conference on Image Processing*, pages 2101–2104, Hong Kong, China, September 2010.

[WSS19] Y. William, S. Safwat, and M. A. M. Salem. Robust Image Forgery Detection Using Point Feature Analysis. In *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 373–380, Leipzig,Germany, September 2019.

[XYS$^+$16] Z. Xia, C. Yuan, X. Sun, D. Sun, and R. Lv. Combining Wavelet Transform and LBP Related Features for Fingerprint Liveness Detection. *IAENG International Journal of Computer Science*, 43(3):290—-298, April 2016.

[ZG13] J. Zhao and J. Guo. Passive Forensics for Copy-Move Image Forgery using a Method based on DCT and SVD. *Forensic Science International*, 233(1):158–166, December 2013.

[Zha19] D. Zhang. *Wavelet Transform*, pages 35–44. Springer International Publishing, May 2019.

[ZKR08] Z. Zhang, J. Kang, and Y. Ren. An Effective Algorithm of Image Splicing Detection. In *Proceedings of the International Conference on Computer Science and Software Engineering*, volume 1, pages 1035–1039, Hubei, China, December 2008.

[ZLLW11] X. Zhao, J. Li, S. Li, and S. Wang. Detecting Digital Image Splicing in Chroma Spaces. In *Proceedings of the 9th International Workshop in Digital Watermarking*, volume 6526, pages 12–22, Seoul, Korea, October 2011.

[ZMM$^+$19] M. Zampoglou, F. Markatopoulou, G. Mercier, D. Touska, E. Apostolidis, S. Papadopoulos, R. Cozien, I. Patras, V. Mezaris, and I. Kompatsiaris. Detecting Tampered Videos with Multimedia Forensics and Deep Learning. In *Proceedings of the International Conference on Multimedia Modeling*, pages 374–386. Springer, 2019.

[ZWLL15] X. Zhao, S. Wang, S. Li, and J. Li. Passive Image-Splicing Detection by a 2-D Noncausal Markov Model. *IEEE Transactions on Circuits and Systems for Video Technology*, 25(2):185–199, February 2015.

[ZWWZ17] Z. Zhang, D. Wang, .C Wang, and X. Zhou. Detecting Copy-Move Forgeries in Images Based on DCT and Main Transfer Vectors. *KSII Transactions on Internet and Information Systems*, 11(9):4567–4587, September 2017.

[ZZ12] Y. Zhang and C. Zhao. Revealing Image Splicing Forgery Using Local Binary Patterns of DCT Coefficients. In *Proceedings of the Communications, Signal Processing, and Systems*, pages 181–189, New York, USA, January 2012.

Parte II

Resumen en Español de la Investigación

Capítulo 9

Introducción

El amplio uso de cámaras digitales en dispositivos móviles es una realidad en la vida cotidiana. Diariamente pueden verse imágenes generadas por dispositivos móviles en telenoticias, correo electrónico o en redes sociales. Webs como Facebook, Youtube o Twitter entre otras, se sitúan en los puestos más altos de la lista de webs más visitadas, siendo una parte considerable de su contenido capturado con cámaras digitales de dispositivos móviles. Todo esto hace que en ciertos casos existan restricciones legales o limitaciones a su utilización en lugares como: colegios, universidades, oficinas de gobierno, empresas, etc. Además, y como consecuencia de todo lo anterior, cada día las imágenes digitales son más utilizadas como testigos silenciosos en procesos judiciales (pornografía infantil, espionaje industrial, violencia callejera, redes sociales), siendo en muchos casos piezas cruciales de la evidencia de un crimen.

Por otro lado, las imágenes digitales se modifican utilizando software de edición especializado que cada vez automatiza y facilita más el trabajo de quien los usa. Tareas como Copiar y Mover una región de una fotografía para ocultar algo o alguien de la escena se puede hacer con muy pocos clics de ratón y con acabados prácticamente imperceptibles para la vista humana. Esto hace que el análisis forense de imágenes digitales sea cada vez más importante.

9.1. Identificación del Problema

Como se ha descrito anteriormente, el hecho de depender de imágenes digitales como fuente potencial de información plantea un problema importante. Los principales factores que contribuyen a esta desconfianza son:

- La existencia de herramientas profesionales para el post-procesamiento de imágenes a bajo coste y disponibles tanto para usuarios principiantes como expertos que permiten crear imágenes falsificadas realistas y muy convincentes.

- Las prácticas cada vez más comunes de intercambio de imágenes por parte de los usuarios de Internet.

Para abordar esta problemática se han desarrollado técnicas forenses que permiten verificar la autenticidad de las imágenes digitales, detectando en su caso manipulaciones en el contenido de las mismas. Estas técnicas se pueden clasificar en dos categorías: activas y pasivas [Con11][FCD12][ZMM+19].

Las técnicas activas de detección de manipulaciones son las más intrusivas ya que se basan en el uso de firmas y marcas de agua que se almacenan en la imagen en el momento de su creación. Se necesita, por tanto, hardware y software especializados para crear y almacenar esta información.

Por el contrario, las técnicas pasivas de detección de manipulaciones no tienen conocimiento previo de la imagen ni de sus características. Estas técnicas utilizan la información del contenido de la imagen para detectar los rastros que dejan las manipulaciones realizadas en ella. Los métodos pasivos tienen, por tanto, una gama más amplia de aplicación y son más útiles para poder detectar y localizar las manipulaciones en las imágenes.

Las técnicas pasivas se clasifican a su vez por el tipo de análisis que realizan: las basadas en el análisis de los metadatos de la imagen, las basadas en el análisis de los píxeles para encontrar patrones que dejen rastros de las manipulaciones y las basadas en el análisis de factores físicos como la luz para identificar inconsistencias en la imagen.

9.2. Motivación

A pesar de que en la literatura se han propuesto muchas técnicas pasivas para validar la autenticidad de las imágenes digitales, la aparición de herramientas de edición cada vez más robustas y de fácil uso crean una necesidad persistente de técnicas forenses de detección de manipulaciones de imágenes más efectivas, que aborden los desafíos que afrontan los analistas forenses actualmente.

Los principales retos que motivan esta tesis son la detección de múltiples regiones clonadas, la localización de áreas clonadas, la robustez de la técnica forense frente a transformaciones geométricas y cómo el ruido y la compresión consecuencia de la manipulación afectan a la imagen resultante.

La investigación de esta tesis parte del estudio de las técnicas existentes en la literatura y se centra en mejorar algunas de las limitaciones existentes en las mismas.

9.3. Objetivos

El objetivo general de esta tesis se centra en la creación, adaptación y mejora de técnicas forenses que agilicen la verificación de la autenticidad de imágenes digitales. Más específicamente, la investigación se enfoca en la detección de manipulaciones del tipo clonación de regiones de una imagen y del tipo empalme de imágenes digitales. Se eligen estos dos tipos de manipulaciones por ser las más utilizadas actualmente. Para alcanzar esta meta se proponen los siguientes objetivos específicos:

1. Investigar las fortalezas y debilidades de las técnicas forenses de detección pasiva de manipulaciones existentes en la literatura.

2. Desarrollar una técnica para clasificar las imágenes manipuladas procedentes de múltiples medios (Internet, redes sociales, USB, discos externos).

3. Analizar y desarrollar técnicas pasivas que localicen manipulaciones del tipo empalme en imágenes digitales.

4. Analizar y desarrollar técnicas pasivas que detecten el clonado y localicen las regiones manipuladas en imágenes digitales.

9.4. Resumen de las Contribuciones

Los resultados de la investigación realizada en esta tesis comprenden diversas contribuciones que han sido publicadas en revistas internacionales indexadas en el Journal Citation Reports y en congresos especializados del área. Como se representa en la Figura 9.1, estas contribuciones se enmarcan en el análisis forense de imágenes digitales. Más específicamente, están centradas en las denominadas técnicas pasivas de detección de manipulaciones de imágenes digitales.

La primera contribución de la tesis, presentada en el Capítulo 4, es una técnica de clasificación automática de imágenes manipuladas. La técnica propuesta combina el análisis de patrones locales de textura y las características de compresión de la imagen con el uso de técnicas de aprendizaje supervisado para verificar la autenticidad de una imagen, identificando manipulaciones del tipo Copiar y Mover y del tipo Empalme [AVSOVHC18] [AVQHSOGV19].

Una vez identificadas las imágenes como manipuladas, las siguientes 4 contribuciones de esta tesis las analizan individualmente para localizar la región objeto de la alteración, especificando 4 técnicas pasivas basadas en el análisis de los píxeles a diferentes niveles de profundidad para encontrar patrones que dejen rastros de las manipulaciones.

Así, la segunda contribución, descrita en el Capítulo 6, consiste en una técnica para la detección de manipulaciones en imágenes digitales que estima el patrón de interpolación utilizado por el sensor de la cámara que generó la imagen, para identificar las inconsistencias presentes en la imagen dejadas por el proceso de edición tras realizar manipulaciones de tipo Copiar y Mover y del tipo Empalme [AVGFSOGV20a].

La tercera y cuarta contribución constituyen el Capítulo 5, donde se describen dos técnicas que detectan manipulaciones de tipo Empalme mediante un mapa de calor. La primera de ellas analiza el nivel de error que resalta los píxeles que tienen un grado de compresión distinto al resto de la imagen original, identificándolos como contenido no original de la imagen. La segunda determina si existe o no modificación en una imagen estimando el patrón de interpolación y el error cuadrático medio [AVMHPA+19] [AVGFSOGV20b].

Finalmente, la quinta contribución, objeto del Capitulo 7, propone una técnica que detecta manipulaciones del tipo Copiar y Mover dentro de una imagen a partir de los patrones estadísticos de los coeficientes de cuantificación de la Transformada Discreta del Coseno [AVMHPA+20].

Todas las técnicas propuestas en esta tesis han sido evaluadas utilizando, además de los datasets de imágenes más utilizados en la literatura, un dataset propio con una variedad de escenarios.

Finalmente, todas estas técnicas se han integrado en una herramienta de análisis forense desarrollada en el marco de un una herramienta de análisis forense, desarrollada dentro de un proyecto del Programa Marco de Investigación e Innovación de la Unión Europea Horizonte 2020, herramienta que ha sido además validada por Policías de diversos países europeos en un entorno real.

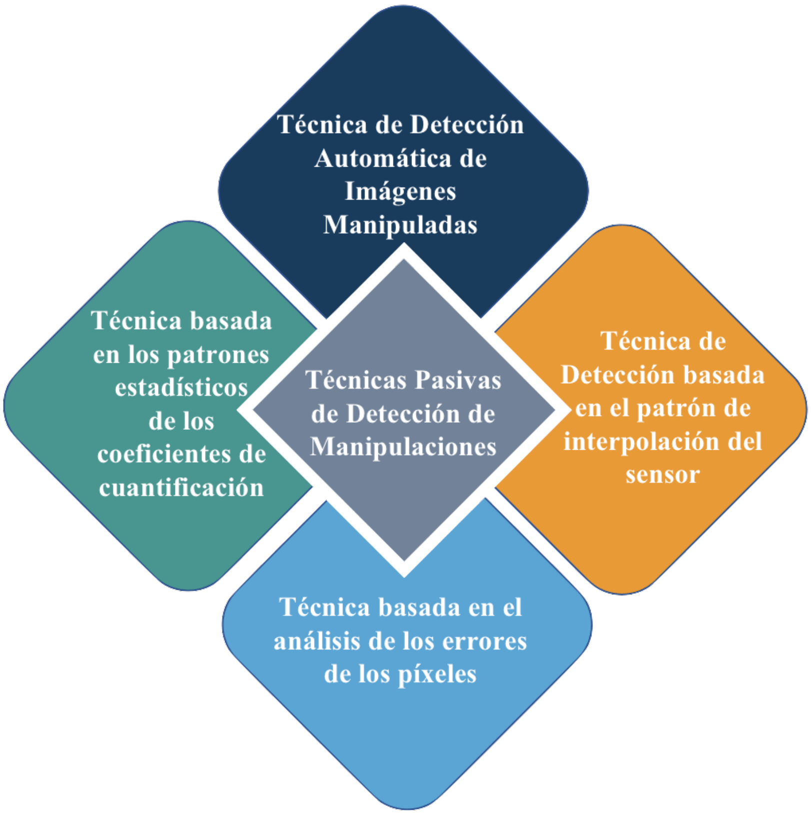

Técnica de Detección Automática de Imágenes Manipuladas
Técnica basada en los patrones estadísticos de los coeficientes de cuantificación
Técnicas Pasivas de Detección de Manipulaciones
Técnica de Detección basada en el patrón de interpolación del sensor
Técnica basada en el análisis de los errores de los píxeles

Figura 9.1: Contribuciones de la tesis

9.5. Estructura del Trabajo

Esta tesis se estructura como sigue:

El Capítulo 2 presenta algunos conceptos básicos sobre imágenes digitales, su proceso de creación y los componentes que allí intervienen. Luego, hace una descripción de los tipos de manipulaciones en imágenes digitales, cuáles fueron sus orígenes y qué técnicas forenses existen, detallando cada una de ellas para comprender mejor las contribuciones de este trabajo.

El Capítulo 3 aborda las técnicas forenses más actuales para identificar manipulaciones en imágenes digitales, explicándolas en profundidad para conocer los componentes que intervienen en el análisis forense de imágenes digitales.

El Capítulo 4 describe la primera contribución de este trabajo, a saber, una técnica de identificación de manipulaciones en imágenes basada en las características extraídas de la transformada wavelet, de la transformada discreta del coseno y del patrón binario local de las imágenes [AVSOVHC18] [AVQHSOGV19].

En el Capítulo 5 especifica la técnica de identificación de manipulaciones en imágenes digitales de tipo Empalme, que estima los patrones de interpolación y consecuentemente permite identificar las anomalías de este patrón presentes en las secciones empalmadas de la imagen manipulada [AVGFSOGV20a].

En el Capítulo 6 desarrolla dos técnicas para identificar y localizar manipulaciones del tipo Copiar y Mover y del tipo Empalme [AVMHPA+19] [AVGFSOGV20b]. La primera de ellas se basa en el error medio cuadrático del patrón de interpolación de la imagen y la segunda en el análisis del nivel de error en la compresión Joint Photographic Experts Group (JPEG) de la imagen.

En el Capítulo 7 muestra una técnica para la detección de manipulaciones del tipo Copiar y Mover. La técnica utiliza las características extraídas de los bloques superpuestos de la imagen utilizando la transformada discreta del coseno [AVMHPA+20].

Finalmente, el Capítulo 8 contiene las principales conclusiones de este trabajo y las líneas futuras de trabajo que del mismo se desprenden.

9.6. Audiencia del Trabajo

Los requisitos previos para acceder al material de esta tesis no son elevados. Se repiten varias definiciones y conceptos disponibles en la literatura con la finalidad de hacer este trabajo independiente. Sin embargo, el lector debe tener presente que es importante tener conocimientos básicos teóricos y prácticos sobre análisis forense de imágenes digitales, cálculos estadísticos, transformada wavelet y algoritmos computacionales. Algunos de los problemas involucrados se discuten aquí, mientras que de otros se considera que el lector ya tiene fundamentos para entender los conceptos discutidos. La bibliografía proporciona al lector información adicional del estudio realizado en esta tesis.

Capítulo 10

Conclusiones y Trabajo Futuro

Este trabajo ha tenido por objeto avanzar en el ámbito del análisis forense de imágenes digitales, haciendo especial hincapié en las denominadas técnicas pasivas que tienen por objeto corroborar la autenticidad de las imágenes digitales, detectando en su caso manipulaciones en el contenido de las mismas. Más específicamente, las técnicas aquí desarrolladas pretenden detectar manipulaciones del tipo Copiar y Mover y del tipo Empalme, por ser las más habituales.

El punto de partida de esta tesis fue la revisión exhaustiva del estado del arte de este tipo de técnicas, analizando si algunas de ellas eran susceptibles de mejora pues se pretendía además que al término de la misma hubiera una herramienta forense de uso policial totalmente operativa. Las contribuciones de esta tesis se resumen en 5 técnicas forenses que se describen seguidamente:

La primera de ellas es una técnica diseñada para ser el punto de partida del análisis forense de imágenes digitales. La técnica propuesta está basada en la combinación de la transformada wavelet con los histogramas extraídos a partir de los patrones binarios locales para la detección de manipulaciones. Esta técnica utiliza como método de clasificación una máquina de soporte vectorial.

Más en detalle, la técnica consiste en la extracción de las características wavelets de la imagen que, posteriormente, son analizadas con el patrón binario local y, finalmente, se calcula el histograma a los bloques resultantes. Luego, y con objeto de mejorar la precisión, se aplica el filtro de espejo en cuadratura, consiguiéndose las características que sirven para entrenar la máquina de soporte vectorial para clasificar posteriormente un conjunto de imágenes entre originales y manipuladas. Para evaluar esta técnica se realizaron varios experimentos con un conjunto de imágenes ampliamente utilizado en el estado del arte. Los resultados mostraron que la técnica obtuvo una precisión máxima del 99,43 % sobre la base de datos CASIA v2.0, mejorando la literatura existente. Además, demostró ser eficiente en las pruebas realizadas tras comparar su tiempo de ejecución con los existentes en el estado del arte. Esta primera contribución ha permitido alcanzar el segundo objetivo planteado al inicio de esta tesis: desarrollar una técnica que de forma automática pueda clasificar las imágenes en manipuladas u originales.

Gracias a su integración con la plataforma RAMSES, las imágenes pueden proceder de una diversidad de fuentes, desde imágenes contenidas en dispositivos de almacenamiento tradicional como discos duros y memorias externas, hasta imágenes procedentes de redes sociales. Sin embargo, dentro de un proceso de análisis forense de imágenes digitales no es suficiente con determinar si es original o si hay indicios de manipulaciones, es necesario además lograr identificar el tipo de manipulación y la región manipulada de la imagen. En este contexto se sitúan las restantes 4 contribuciones de esta tesis que logran los objetivos de identificar manipulaciones del tipo Empalme y del tipo Copiar y Mover, localizando además la región manipulada dentro de la imagen.

En cuanto a la identificación y localización de manipulaciones de tipo Empalme se han desarrollado tres técnicas forenses. Dos de estas tres técnicas se basan en el análisis de las huellas que dejan las manipulaciones en imágenes con compresión JPEG. Más concretamente, una de ellas se basa en las características que deja la compresión con pérdida JPEG utilizando la técnica de análisis de nivel de error que es capaz de resaltar los píxeles que contienen un nivel de compresión distinto al del resto de la imagen y en base a esto se marcan las áreas que no pertenecen al contenido original. La segunda detecta manipulaciones en una imagen en color utilizando algoritmos de interpolación cromática de la siguiente forma: divide la imagen en bloques solapados y estima el patrón de interpolación y el error cuadrático medio de estos bloques, analizando las anomalías encontradas e identificando si existe o no manipulación en la imagen a nivel de píxel, permitiendo de esta forma localizar la región manipulada.

Estas dos técnicas fueron evaluadas con conjuntos de datos públicos y ampliamente utilizados en el estado del arte. Los experimentos realizados para la primera técnica utilizaron la base de datos CASIA v1.0 así como un conjunto de datos propio, demostrándose que la calidad de la imagen incide directamente en los resultados porque en el caso de CASIA, que cuenta con una baja calidad de resolución, la técnica encontró algunas dificultades para detectar la región que no pertenece a la imagen original. Sin embargo, en el caso del conjunto de datos propio que contiene imágenes de alta resolución, la técnica alcanzó una precisión del 73,3 % de acierto. Además, gracias a la rapidez de esta técnica, puede ser muy útil para el analista forense a la hora de llevar a cabo análisis de grandes cantidades de imágenes, pudiendo ser un primer filtro para continuar analizando las imágenes detectadas como manipuladas con técnicas más sofisticadas y robustas, pero bastante más lentas.

Los experimentos realizados para la segunda técnica fueron satisfactorios al introducir imágenes con dimensiones superiores a 1500x1500 píxeles. Sin embargo, las imágenes donde hay escenas con regiones de colores blancos muy grandes generan falsos positivos debido a que la varianza calculada en esas secciones es muy baja con respecto al resto de la imagen. Por otro lado, al analizar imágenes con resoluciones menores o iguales a 700x700 píxeles, la técnica tiene dificultades para detectar el área con modificaciones, ya que la información de la imagen no es suficiente para hacer la diferencia entre las varianzas conocidas. A pesar de ello, esta técnica, como en el caso anterior, puede servir como un primer filtro para separar de un gran número de imágenes aquellas más susceptibles de haber sido manipuladas y, posteriormente, utilizar otra técnica más precisa que realice un análisis más exhaustivo.

La tercera técnica identifica manipulaciones, tanto de tipo Empalme como de tipo Copiar y Mover, debido a que utiliza las marcas que deja la matriz de filtro de color de la cámara en el momento de generar la imagen capturada. La técnica permite generalizar la detección de manipulaciones a distintos filtros de color sin perder precisión, como sucede con técnicas similares que trabajan en el espacio de color RGB. Esta técnica puede ser aplicada en diversos escenarios siempre que se presenten marcas de forma periódica en la imagen analizada. Por su configuración utiliza diferentes bandas y un menor tamaño de bloque que los propuestos en la literatura, permitiendo mejorar la precisión a la hora de identificar la región manipulada. Además, al utilizar otras bandas de color distintas a la verde, permite identificar manipulaciones a pesar de que a la imagen se le haya aplicado algún filtro después de ser manipulada. Los experimentos realizados para validar esta técnica utilizaron el conjunto de datos RAISE y el conjunto de datos CMFD. La cuarta y última técnica identifica manipulaciones del tipo Copiar y Mover y, tras realizar un análisis de bloques de la imagen, permite localizar la región manipulada con una precisión promedio del 97,88 %. El conjunto de datos utilizado consta de un total de 1492 imágenes con diferentes resoluciones y pos-procesamientos. Las imágenes analizadas que contenían un pos-procesamiento fueron las que dieron una mayor cantidad de falsos positivos y falsos negativos. Sin embargo, la técnica permitió identificar manipulaciones tras la aplicación de filtros de color y transformaciones geométricas leves (rotaciones, cambios de tamaño, etc.).

10.1. Trabajo Futuro

El análisis forense de imágenes digitales y, en especial, la detección de falsificación de imágenes es, sin duda, una disciplina de investigación amplia y actual. A medida que las aplicaciones de edición de imágenes son cada vez más sofisticadas y fáciles de utilizar, es más fácil crear falsificaciones más realistas y convincentes. Asimismo, a medida que las técnicas de detección de falsificaciones se vuelven más conocidas se irán descubriendo vulnerabilidades que serán explotadas para robustecer las técnicas de falsificación. Por todo lo anterior y, aunque las técnicas forenses pasivas de detección de manipulaciones de imágenes digitales propuestas en esta tesis han logrado resultados satisfactorios, queda mucho por hacer. A continuación, se resumen algunas potenciales líneas futuras de investigación:

- **Ampliar los tipos de manipulaciones detectadas**: Las técnicas desarrolladas en esta tesis están basadas en el análisis de la compresión con pérdida (Capítulo

5) y en el análisis de píxeles (Capítulos 4, 6, 7 y 8), con el objetivo de detectar manipulaciones de los tipos Copiar – Mover y Empalme. Se pueden combinar estas técnicas con cualquiera de las otras técnicas pasivas presentadas en el Capítulo 3 para manipulaciones del tipo Re-Muestreo y del tipo Falsificación de la Fuente de Adquisición. Este tipo de combinación podría aumentar significativamente los resultados obtenidos gracias a la diversidad de las técnicas utilizadas.

- **Extender las técnicas detección de manipulaciones del tipo Copiar – Mover y Empalme a vídeos digitales**: A medida que aumenta el interés en el análisis forense pasivo de imágenes, la comunidad científica también se muestra interesada en la aplicación y extensión de estas técnicas a la problemática de las falsificaciones de vídeos. El análisis fotograma a fotograma de un vídeo se convierte así en la aplicación inmediata; sin embargo, la naturaleza temporal de los vídeos añade una complejidad adicional y abre nuevas vías de estudio.

- **Aplicar técnicas de aprendizaje profundo a la detección automática de manipulaciones**: Recientemente, las técnicas de aprendizaje profundo han demostrado ser efectivas y con un rendimiento notable en aplicaciones de inteligencia artificial y de visión por ordenador. Sería interesante ejecutar y comparar la técnica de detección automática de manipulaciones del Capítulo 4, que utiliza técnicas de aprendizaje supervisado con técnicas de aprendizaje profundo aumentando además el tamaño de los conjuntos de datos de entrenamiento utilizados.

- **Mejorar la efectividad y robustez de las técnicas basadas en el análisis de la compresión JPEG con el uso de filtros automáticos de eliminación de ruido**: Las principales limitaciones de la técnica de análisis de nivel de error, desarrolladas en el Capítulo 5, son su limitada aplicación a imágenes sometidas a compresión con pérdida y que la decisión final la toma el analista forense al interpretar los resultados de la detección. Por tanto, incluir un filtro adaptativo de eliminación del ruido generado por el esquema de compresión utilizado podría mejorar el nivel de error. Atenuar con éxito este ruido mejoraría los niveles de error, identificando mejor las regiones de la imagen donde se ha producido la manipulación. Con esto se facilitaría la tarea del analista forense.

- **Explorar otras características adicionales presentes en las imágenes para identificar las manipulaciones con más detalle**: Aunque las técnicas de detección de manipulaciones del tipo Copiar y Mover de los Capítulos 6 y 7 proporcionan muy buenos resultados, se puede estudiar el uso de las técnicas de extracción de descriptores y características locales invariantes para proporcionar una localización más precisa de las regiones manipuladas en imágenes con baja frecuencia espacial (por ejemplo regiones lisas con textura mínima).

Part III

Papers Related to This Thesis

Chapter 11

List of Papers

Below are the works that are derived to the accomplishment of the present Doctoral Thesis:

1. Esteban Alejandro Armas Vega, Edgar González Fernández, Ana Lucila Sandoval Orozco, Luis Javier García Villalba. Passive Image Forgery Detection Based on the Demosaicing Algorithm and JPEG Compression. IEEE Access, 8(1):11815–11823, January 2020.

2. Esteban Alejandro Armas Vega, Edgar González Fernández, Ana Lucila Sandoval Orozco, Luis Javier García Villalba. Image tampering detection by estimating interpolation patterns. Future Generation Computer Systems, 107:229–237, June 2020.

3. Esteban Alejandro Armas Vega, Luis Alberto Martínez Hernández, Sandra Pérez Arteaga, Ana Lucila Sandoval Orozco, Luis Javier García Villalba. Detección de Manipulaciones Copy-Move en Ficheros Multimedia mediante la Transformada Discreta del Coseno. Actas de el X Congreso Iberoamericano de Seguridad Informática (CIBSI 2020), Bogotá, Colombia, Enero 21 – 24, 2020.

4. Esteban Alejandro Armas Vega, Luis Alberto Martínez Hernández, Sandra Pérez Arteaga, Ana Lucila Sandoval Orozco, Luis Javier García Villalba. Algoritmo de Interpolación Cromática para la Detección de Zonas Manipuladas de Imágenes. Actas de las V Jornadas Nacionales de Ciberseguridad (JNIC 2019), pages 200 – 205, Cáceres, España, Junio 5 – 7, 2019.

5. Esteban Alejandro Armas Vega, Carlos Quinto Huamán, Ana Lucila Sandoval Orozco, Luis Javier García Villalba. Técnica de Autenticación de Imágenes Digitales Basada en la Extracción de Características. Actas de las V Jornadas Nacionales de Ciberseguridad (JNIC 2019), pages 291 – 296, Cáceres, España, Junio 5 – 7, 2019.

6. Esteban Alejandro Armas Vega, Ana Lucila Sandoval Orozco, Luis Javier García Villalba, Julio César Hernández-Castro. Digital Images Authentication Technique Based on DWT, DCT and Local Binary Patterns. Sensors, 18(10):3372, October 2018.