Evaluación y comparación de un grupo de técnicas para la identificación de imágenes digitales alteradas por remuestreo

Valentina Pareja Rúa

Proyecto de grado para optar al título de
Ingeniera Física

Universidad Tecnológica de Pereira
Facultad de Ingenierías
Ingeniería Física
Pereira, Colombia
2019

Evaluación y comparación de un grupo de técnicas para la identificación de imágenes digitales alteradas por remuestreo

Por:
Valentina Pareja Rúa
Cód: 1093226944

Proyecto de grado para optar al título de
Ingeniera Física

Director:

M.Sc. Jimmy Alexander Cortés Osorio
Profesor Titular del Departamento de Física
Ingeniero Electricista

Universidad Tecnológica de Pereira
Facultad de Ingenierías
Ingeniería Física
Pereira, Colombia
2019

Índice general

Capítulo 1

Introducción

La falsificación de imágenes digitales, en la actualidad, es una tarea que se realiza con mucha facilidad debido al avance de la tecnológica y a la aparición de softwares como Photoshop y Corel Draw. Es común encontrar imágenes con alteraciones en el mundo de la moda y la farándula pero cuando se trata de procesos legales o noticias puede traer consecuencias negativas a los implicados. Existen diferentes métodos de falsificación de imágenes, uno de ellos es el remuestreo o resampling, el cual consiste en copiar y pegar una porción de la imagen en si misma y realizar una transformación geométrica como la rotación y/o escalado. El resampling introduce nuevos píxeles a partir de los existentes, por medio de una interpolación, este proceso crea una correlación con una forma especifica entre los píxeles y altera la estadística de la imagen, lo cual no ocurre de forma natural [1]. Una imagen puede contener remuestreo pero no necesariamente contiene una alteración maliciosa, ya que el simple hecho de ampliar o cambiar de tamaño la imagen introduce correlaciones. Las técnicas para detectar el remuestreo buscan la correlación entre píxeles y/o alteraciones en la estadística de la imagen. Los métodos consisten en un preprocesamiento, extracción de características y clasificación. En el preprocesamiento se encuentran cambios de escala para reducir la dimencionalidad de la imagen y el tiempo de computo. En la extracción de características se encuentran filtrados para resaltar detalles, extracción de puntos claves invariantes a la rotación y escalado, transformadas de Radon y DCT (Discrete Cosine Transform), y análisis estadísticos. La mayoría de estos métodos es complementado con la transformada de Fourier con el fin de que la periodicidad de las correlaciones sean más evidentes. La clasificación de una imagen en auténtica o manipulada se convierte en un problema de reconocimiento de patrones de dos clases, para ello se usa aprendizaje de maquina, como las máquinas de soporte vectorial (SVM) y umbrales para detectar picos en el dominio de la frecuencia. La parecencia de picos dominantes es un indicador de la presencia del remuestreo. De acuerdo a lo anterior, se seleccionará un grupo de cuatro técnicas de la literatura científica para evaluar la exactitud, sensibilidad, especificidad entre otras métricas y así poder identificar las restricciones que posee cada una de ellas. Estas técnicas serán evaluadas en

bases de datos reconocidas de la literatura, que contengan este tipo de falsificación, con el propósito de observar el efecto que pueden llegar a causar las diferentes características de las imágenes en cada una.

Capítulo 2

Justificación

El área de la informática forense ha ido en aumento en los últimos años debido a la importancia que se le da a la información digital y a su uso masivo. La informática forense es un apoyo a los procesos de la justicia para enfrentar los retos impuestos por el avance de la tecnología. La alteración de imágenes es algo común en el entretenimiento, sin embargo, cuando se trata con imágenes que hacen parte de un proceso jurídico es necesario el uso de la informática forense para asegurar la validez de la imagen. En el código penal colombiano [2] las imágenes cuentan como material probatorio de un proceso jurídico. El gobierno de Colombia busca estandarizar la manipulación de evidencias digitales halladas en equipos electrónicos. En el documento "Seguridad y privacidad de la información, evidencia digital" [3] y por medio de la publicación "Mintic busca oferta académica para formación en áreas de gestión de TI y seguridad y privacidad de la información" [4] la república de Colombia expresa su necesidad por tecnologías que faciliten la custodia y el mantenimiento de la validez jurídica de la evidencia digital como las imágenes. El país no cuenta con una herramienta oficial que permita certificar la veracidad de las imágenes. Las noticias falsas son frecuentes en las redes sociales, se aprovechan de la credibilidad que tienen a las imágenes a la hora de validar un hecho, por ende, personas ingenuas pueden ser engañadas. De allí se hace necesario identificar si una imagen es falsa en una red social. Un primer acercamiento a la solución de estas problemáticas es identificar un método exacto para la detección de remuestreo en imágenes digitales. En la actualidad existen diversos métodos para la detección de falsificaciones en imágenes basados en el tipo de alteración de remuestreo, pero se desconoce que técnica es más exacta. Además, estas técnicas han sido evaluadas con bases de datos diferentes para cada una, lo que hace difícil la comparación entre ellas debido a las diversas características de las imágenes.

En la Universidad Tecnológica de Pereira el Grupo de Investigación de Robótica Aplicada (GIRA) esta elaborando un proyecto financiado por la vicerrectoría de investigación e innovación en la línea de visión artificial titulado "Propuesta metodológica para la identificación de imágenes digi-

tales alteradas por copy-move, resampling y splicing", con código 3188 del cual este proyecto hace parte.

2.1. OBJETIVOS

2.1.1. Objetivo General

Evaluar y comparar un grupo de técnicas para la detección de la manipulación de tipo remuestreo en una imagen digital y establecer las limitaciones que pueden llegar a tener estas técnicas debido al ruido y compresión JPEG.

2.1.2. Objetivos Específicos

1. Seleccionar un grupo de cuatro técnicas que pertenezcan al dominio del espacio y la frecuencia para identificar la falsificación en imágenes de tipo remuestreo.

2. Implementar y evaluar el grupo seleccionado de cuatro técnicas con un conjunto de imágenes de una base de datos reconocida.

3. Comparar los resultados obtenidos mediante técnicas de validez para estimar su exactitud en la identificación de imágenes alteradas por remuestreo.

Capítulo 3

Estado del Arte

Esta investigación se desarrollara en torno a la técnica basada en píxeles para la detección de la falsificación del tipo remuestreo.

Tong Qiao, Aichun Zhu y Florent Retraint (2018) [5] investigaron el problema de la detección de remuestreo de imágenes basada en el modelo paramétrico lineal. Primero, estudian el artefacto periódico de una señal remuestreada unidimensional 1-D. Después de lidiar con los parámetros molestos, junto con la regla de Bayes, diseñan el detector en función de la probabilidad de ruido residual extraído de la señal remuestreada usando un modelo paramétrico lineal. Posteriormente, estudian la característica de una imagen remuestreada. Proponen estimar la probabilidad de ruido y establecer una relación de verosimilitud (LRT - Likelihood Ratio Test). Este detector puede resolver el problema de autenticar las imágenes remuestreadas comprimidas y sin comprimir mezcladas.

Gajanan K. Birajdar y Vijay H. Mankar (2013-2014) [6] proponen un método para detectar el reescalado y estimar su factor, en función de las propiedades de los cruces por cero de la segunda diferencia de la imagen manipulada . El algoritmo consta de un pre-procesamiento donde se convierte la imagen en escala de grises y se extrae la componente Y después de la conversión a YCbCr, se calcula la segunda diferencia del componente Y, se encuentran los cruces por cero de la secuencia obtenida, se construye una secuencia binaria y se calcula la DFT (Discrete Fourier Transform) de esta secuencia, para detectar la periodicidad, todas las magnitudes son combinadas y graficadas juntas, la deteccion de los picos se hace automáticamente con un umbral simple que busca el máximo local. En el experimento, tomaron 20 imágenes a color sin comprimir de tamaño 384×512 de la base de datos UCID (Uncompressed Color Image Database), la base de datos consiste en 800 imágenes con diferentes factores de escalamiento, el algoritmo fue aplicado a las columnas. En un artículo [7] posterior estos mismos autores retoman el algoritmo, usan la base de datos UCID y USC-SIPI, adicionan ataque como la compresión JPEG. La técnica es robusta y

detecta con éxito la operación de reescalado para imágenes que han sido sometidas a varias formas de ataques como compresión JPEG, se observa cierta degradación en la precisión de detección a medida que aumenta la calidad de la calidad JPEG.

Gajanan K. Birajdar y Vijay H. Mankar (2018) [8] desarrollaron un algoritmo para detectar la operación de escalamiento global de la imagen alterada obteniendo características del submuestreo. El criterio de Fisher se emplea para elegir las características relevantes y reducir la dimensionalidad de las características estadísticas. Se utiliza para la clasificación, máquinas de soporte vectorial (SVM) y una red neuronal artificial (ANN) multi-capa. Los resultados experimentales utilizan imágenes de Cb, Cr y escala de grises, el método propuesto tiene un buen rendimiento de detección de reescalado incluso cuando se trata de distorsiones como la compresión JPEG. Los resultados indican que SVM se desempeña mejor en comparación con el clasificador ANN.

Belhassen Bayar y Matthew C. Stamm. (Department of Electrical and Computer Engineering, Drexel University, Philadelphia, USA, 2017) [9] proponen un método basado en redes neuronales convolucionales (CNN) para detectar el remuestreo en imágenes re-comprimidas. Además, usan una nueva capa llamada «capa convolucional limitada», ha sido diseñada principalmente para detectar falsificaciones de imágenes. Las CNN tiene la propiedad de aprender las características de aprendizaje directamente de los datos. La exactitud del método propuesto disminuye cuándo se usa un factor de calidad bajo para la post-compresión, el enfoque propuesto puede detectar hasta un factor de post-compresión de 50.

Zhipeng Chen, Yao Zhao y Rongrong Ni (Beijing Jiaotong University, Beijing, China, 2017) [10] abordaron el tema de las cadenas de operaciones en imágenes falsificadas, específicamente analizan las cadenas de operaciones de compresión JPEG (JP) y remuestreo (RS) con escalamiento (RE) y rotacion (RO). Las cadenas de operaciones analizadas son las siguientes, para JP-RS se tiene las siguientes combinaciones JP-RE, JP-RO, JP-RE-RO y JP-RO-RE; para JP-RS-JP las combinaciones son JP-RO-JP, JP-RE-RO-JP y JP-RO-RE-JP. Caracterizan la cadena de operaciones mediante los artefactos de bloques transformados (TBAG - transformed block artifact grid), se analizan tanto en el dominio de píxel como en el de las transformaciones de coseno discreto (DCT) y se utilizan para diseñar un esquema de detección. El esquema de detección tiene el siguiente orden: dada una imagen de prueba como entrada, el primer paso es juzgar si la imagen de prueba ha sufrido la compresión JPEG como la última operación. Cuando la compresión JPEG es la última operación, el residuo de la transformada discreta de coseno (DCTR - Discrete Cosine Transform Residual) se usa para distinguir entre imágenes sometidas a JP-RS-JP o solo a JP por un clasificador de máquinas de soporte vectorial (SVM). Si la compresión JPEG no es la última operación como

se juzgó en la primera etapa, entonces se extraen las características de TBAG para distinguir entre las imágenes manipuladas por JP-RS o alguna otra operación única, igualmente se usa un clasificador de máquinas de soporte vectorial (SVM). Para las características TBAG se usa un detector que consiste en cuatro pasos, primero calculan las diferencias entre pixeles, segundo detectan y eliminan bordes, tercero aplican la transformada rápida de Fourier (FFT - Fast Fourier Transform) y eliminan el ruido, y cuarto extraen las características. Para el DCTR las características que se capturarán son las estadísticas derivadas de los coeficientes DCT interbloque. Específicamente, las matrices de DCT no determinadas que se calculan utilizando 64 convolucionales con 64 patrones de base de DCT. La función DCTR se extrae de los histogramas de las matrices de DCT no determinadas. La función DCTR es capaz de caracterizar las características híbridas de los coeficientes DCT resultantes de JP-RS-JP. El análisis se realiza sabre la capa de luminiscencia de la imagen. En este artículo no se considera la reducción de tamaño de la imagen (down-samplig).

Jason Bunk et al (USA, 2017) [11] proponen dos métodos para detectar y localizar manipulaciones de imágenes basadas en una combinación de características de remuestreo y aprendizaje profundo. En el primer método, la transformada de Radon de las características de remuestreo se computa en bloques de imagen superpuestos. Los clasificadores de aprendizaje profundo y un modelo de campo aleatorio condicional gaussiano se utilizan para crear un heatmap. Las regiones manipuladas se ubican utilizando un método de segmentación de Randon Walker. En el segundo método, las características de remuestreo calculadas en parches de imagen superpuestos se pasan a través de una red basada en memoria a largo plazo (LSTM - Long short-term memory) para clasificación y localización. Comparan el rendimiento de detección y localización de estos dos métodos. los resultados experimentales muestran que ambas técnicas son efectivas para detectar y localizar falsificaciones de imágenes digitales.

Yuting Su, Xiao Jin, Chengqian Zhang, Yawei Chen (School of Electronic Information Engineering, Tianjin University, Tianjin, China, 2017) [12] presentaron un algoritmo para detectar el remuestreo mediante la deconvolución ciega, recuperan las imágenes editadas usando un proceso de filtrado inverso. Puede detectar diferentes tipos de interpolación en el proceso de remuestreo y evitar la interferencia causada por artefactos de bloque JPEG.

Alin C. Popescu y Hany Farid (Computer Science Department, Dartmouth College, Hanover, 2005) [1] expusieron un método que consiste en un algoritmo llamado Expectation/Maximization (EM), sirve para estimar un conjunto de muestras periódicas que están correlacionadas con sus vecinos, y la forma específica de estas correlaciones. El algoritmo EM es un algoritmo iterativo de dos pasos: 1) en el paso E estima la probabilidad de que el píxel pertenezca a un modelo M1 o M2

y 2) en el paso M se estima la forma especifica de la correlación entre las muestras. la probabilidad de cada muestra que pertenece al modelo M1 se puede obtener usando la regla de Bayes. El algoritmo EM es usado para crear un mapa de probabilidad (p-map). Para notar la periodicidad que introduce el remuestreo en la imagen se hace la transformada discreta de Fourier, de tal forma que una imagen con remuestreo presentara alteraciones en su espectro.

Hieu Cuong Nguyen y Stefan Katzenbeisser (Computer Science Department, Darmstadt University of Technology, Germany, 2012) [13] revisaron las técnicas existentes para detectar el remuestreo, diseñan algunos ataques dirigidos para evaluar la confiabilidad de dichas tecnicas y proponen una técnica mejorada para detectar el remuestreo. La técnica se basa en el cálculo de un pseudo mapa de probabilidad (pseudo p-map) de la imagen a probar y la aplicación de la transformada de Radon a este mapa. Para detectar la periodicidad se hace DFT y se buscan los picos fuertes mediante el cálculo de los máximos locales del espectro seleccionando los picos en función de un umbral predefinido. Dado que la técnica propuesta no necesita la estimación del algoritmo EM (Expectation/Maximization) para calcular el pseudo p-mapa, es mucho más rápida que la propuesta por Popescu y Farid. Segun los resultados, la técnica de Popescu y Farid es más poderosa que la propuesta en [13]. Sin embargo, la técnica propuesta es más robusta cuando se aplican ataques, como la compresión o ruido, para ocultar el remuestreo.

Hieu Cuong Nguyen (Faculty of Technology, University of Transport and Communications, Hanoi, Vietnam, 2016) [14] plantearon un método basado en el trabajo de Popescu y Hany Farid, el cual calcula el mapa de probabilidades de la imagen con un conjunto de valores predeterminados para α (pseudo p-map), aplican la transformada de Radón al pseudo p-map, y se extraen las características especificas del mapa transformado para la clasificación de imágenes. Con el fin de clasificar las imágenes se usan máquinas de soporte vectorial (SVM). Los resultados experimentales muestran que la técnica funciona para imágenes con un factor de escalado mayor a 1.1 y ángulos de rotación mayores a cinco grados.

Anjie Peng, Hui Zeng, Xiaodan Lin y Xiangui Kang (Sun Yat-Sen University, Guangzhou, China, 2015) [15] proponen un método basado en la característica de correlación parcial (PAF) el cual se aplica en las filas y las columnas. Para probar el método que proponen usan la base de datos UCID que contiene 1338 imágenes en escala de grises de tamaño 384×512. Se emplea máquinas de soporte vectorial con núcleo gaussiano, como clasificador y comparan sus resultados con el método propuesto por Popescu y Farid en el 2005.

David Vázquez-Padín y Fernando Pérez-González (Signal Theory and Communications Dept,

University of Vigo, Vigo, Spaña, 2011) [16] combinaron dos técnicas para la detección de una región manipulada. La primera técnica es SIFT (Scale-Invariant Feature Transform), que detecta la duplicación de regiones proporcionando puntos claves de las regiones involucradas en la manipulación y también los parámetros de la transformación geométrica entre ambas regiones. Sin embargo, no hay información sobre cuáles de las regiones son originales y cuáles son las duplicadas, para solucionar este problema proponen un segundo método basado en remuestreo para proporcionar una manera precisa de distinguir las regiones originales y las manipuladas analizando el factor de remuestreo de cada área duplicada, el método basado en remuestreo por si solo presenta una alta tasa de falsos positivos. Los métodos se aplican a la imagen de prueba separadamente, primero se aplica SIFT con un umbral de 0.6 y luego un método basado en remuestreo. la imagen de prueba es dividida en bloques de 128×128. El método esta diseñado para imágenes que no tengan compresión JPEG.

David Vázquez-Padín, Pedro Comesaña y Fernando Pérez-Gonzáles (Signal Theory and Communications Department, University of Vigo, Vigo, Spain, 2015) [17] estudiaron las dependencias locales lineales inducidas por transformaciones espaciales interpoladas linealmente, para ello caracterizan las dependencias lineales a través de la Descomposición del Valor Singular (SVD) de una imagen remuestreada. el método propuesto se basa en el cálculo del SVD de un bloque con tamaño pequeño a partir de una imagen. Este método no esta diseñado para imágenes escaladas a un tamaño más pequeño que el original. la magnitud de los valores singulares de la imagen remuestreada cae más bruscamente que la proviene de la imagen no remuestreada. La presencia de saturación y la posibilidad de tener alguna dependencia lineal entre muestras afectará la evolución esperada de los valores singulares del bloque de imagen, produciendo dos posibles resultados: 1) El número de valores singulares distintos de cero es sustancialmente más pequeño que N (tamaño del bloque). Esto ocurre raramente a menos que varias filas o columnas del bloque estén completamente saturadas. 2) El número de valores singulares distintos de cero está cerca de N, pero su magnitud desaparece más bruscamente de lo normal. Esto ocurre cuando las dependencias lineales están presentes entre las filas o columnas del bloque, y pueden ser aumentadas por saturaciones.

Xiaodan Hou et al. (Henan, China, 2014) [18] presentaron un método para la detección de remuestreo para ello usan el análisis de textura. la influencia de las operaciones de remuestreo en una imagen cruda de muestra única se ve como una alteración de la textura de la imagen en una escala fina. Primero, utilizan una transformada lineal local para obtener subbandas de detalles de textura. A continuación, extraen un vector de características 36-D de los momentos de la función característica normalizada de las subbandas de detalles de texturas para entrenar un clasificador de máquinas de soporte vectorial. Además, experimentos en imágenes de mapa de Stego ilustran que

el método propuesto es esencial para construir estrategias de esteganálisis precisos y ciegos para imágenes heterogéneas e imágenes remuestreadas a diferentes escalas.

Ruohan Qian, Weihai Li, Nenghai Yu y Zhuo Hao (Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, Anhui, China, 2012) [19] formularon una técnica tolerante a la rotación, primero se calcula la derivada de segundo orden de la imagen y se halla el espectro de la energía, si el espectro presenta una periodicidad quiere decir que la imagen tiene remuestreo. A este algoritmo le llaman Rotation Tolerant Resampling Detection (RTRSD). Para localizar la zona que ha sido alterada se procede ha dividir la imagen y a cada sub-imagen se le calcula el espectro de energía de segundo orden con RTRSD, se extrae la posición de los picos de cada espectro de energía, se calcula la distancia entre las sub-imagenes en términos de la tasa patrón de remuestreo. La precisión del algoritmo para determinar si una imagen contiene remuestreo fu del 97.2 % y la exactitud del algoritmo para localizar la zona fue del 93,6 %.

Xiaoying Feng, Ingemar J. Cox y Gwenaël Doërr (Department of Computer Science, University College London, London, 2012) [20] desarrollaron un método basado en la densidad de energía normalizado. Se toma la segunda derivada a lo largo de la dimensión horizontal o vertical en una imagen, posteriormente se aplica la transformada de Radón para proyectar la segunda deriva en 180 direcciones. La detección del remuestreo se basa en localizar la periodicidad en la autocovarianza de los vectores proyectados. Para detectar la autocovarianza se hace uso de la transformada de Fourier discreta, usando un detector de máximos local. Se extrae un vector de características 19-D tras examinar la energía normalizada presente en varios tamaños de ventanas y se usa para entrenar un clasificador de máquinas de soporte vectorial. Esta técnica es significativamente mejor para el down-sampling en comparación con el método de Popescu y Farid, además, el método es deficiente para compresión JPEG y adición de ruido blanco Gausiano.

Stefan Pfennig y Matthias Kirchner (Roma, Italia, 2012) [21] combinaron modelos analíticos de artefactos de interpolación periódica con la distribución de energía espectral de imágenes reescaladas para obtener parámetros de transformación en un entorno forense ciego. En general combinan el método de los mapas de probabilidad con el análisis espectral, utilizando esta información para entrenar un clasificador de máquinas de soporte vectorial. Además por medio de esta combinación hallan el factor de escalado de la imagen.

Yang Ta Kao, Hwei Jen Lin, Chun Wei Wang, y Yi Chun Pai (Tamkang University, Taipei, Taiwan, 2012) [22] propusieron un método para detectar el up-sampling. El método incluye tres partes. Primero, presenta un algoritmo para la construcción de la matriz de remuestreo, que deriva

automáticamente la matriz de muestreo para cualquier factor de muestreo dado. En segundo lugar, mostramos un algoritmo, Zeroing Mask Derivation, que construye una máscara de cero para el remuestreo producido por el algoritmo propuesto. Por último, plantearon un algoritmo para detección de up-sampling, que detecta el remuestreo ascendente en imágenes utilizando Zeroing Mask Derivation en un orden específico.

Juxian Zuo et al (Guizhou University, Guiyang, China, 2011) [23] desarrollaron un algoritmo en el cual primero se divide una imagen posiblemente falsificada en bloques superpuestos, luego definen y extraen un factor de medida de bloque que contiene tanto las características de remuestreo como las características de compresión JPEG para cada bloque, para ello aplican en dirección vertical o horizontal la segunda derivada de la fila o columna según corresponda, se calcula la transformada discreta de Fourier (DFT) en una dimensión y se toma el promedio para obtener el espectro. Por ultimo aplican el factor de medida de bloque para discriminar las regiones manipuladas y regiones no manipuladas. Para reconocer una imagen falsificada automáticamente usan un algoritmo iterativo de segmentación de umbral para dividir el mapa de medida de bloque en dos partes, luego reconocen si se trata de una imagen falsificada basada en la diferencia entre el valor medio de las dos partes. Las ventajas del algoritmo propuesto es que puede detectar imágenes falsificadas con compresión. Otra ventaja es que es eficaz cuando las imágenes compuestas se guardan con un factor de calidad inferior al de la imagen original. Dado que el método se basa en los rastros de remuestreo y compresión JPEG, se vuelve ineficaz cuando la imagen falsificada se compone de una imagen sin comprimir y, mientras tanto, la región manipulada no experimente ninguna operación geométrica.

Geun-Sil Song, Yong-In Yun, y Won-Hyung Lee (2011) [24] presentaron un método el cual usa derivadas de segundo orden y calculan la DFT en las direcciones vertical horizontal y diagonal, Emplearon la detección de picos basada en el umbral para la correlación del espectro DFT. El algoritmo consta de cinco pasos: 1) convierte la imagen a escala de grises, se aplica un filtro pasa bajos para detectar la región interpolada en la imagen falsificada, y eliminar el ruido de la imagen. 2) Se estimaron los factores de interpolación para la dirección horizontal, vertical y diagonal, respectivamente, para ello usan un kernel derivativo en cada dirección. 3) Se calculaó un promedio para las señales de la segunda derivada de cada dirección, los valores absolutos de cada segunda derivada se promedian juntos para obtener la media de la traza. 4) Se buscan las propiedades periódicas del remuestreo utilizando la DFT normalizada en el promedio de la segunda derivada 5) para medir automáticamente el factor de interpolación, emplean la detección de picos basada en umbrales para la correlación del espectro DFT. El método presentado ha sido aplicado a imágenes del tamaño de 512×512 píxeles, presentando buenos resultados para el up-sampling y down-sampling.

Shu-ping Li et al (College of Software Nankai University Tianjin, China, 2010) [25] proponen un método para detectar el remuestreo el cual es capaz de distinguir entre los artefactos periódicos introducidos por el remuestreo y la compresión JPEG. Primero calcularon el mapa de probabilidad (P-map) por me dio del algoritmo EM y se la aplicó la FFT. Si la imagen tiene compresión JPEG se crean múltiples picos de frecuencia de $1/8$ en el dominio de Fourier. Para dar solución a esta problemática introducen un tercer paso al algoritmo original de Popescu y Farid el cual consiste en hacer coincidir la transformada de Fourier del mapa de probabilidad y una plantilla JPEG, si el valor es inferior a un umbral, significa que la imagen tiene remuestreo, de lo contrario, se ha comprimido en JPEG. La tasa de error aumenta significativamente hasta que la calidad disminuye a 80. Con la disminución de la calidad de compresión JPEG, los resultados empeoran y la tasa de error aumenta significativamente. Cuando la compresión JPEG tiene una calidad inferior a 75, es incapaz de detectar si la imagen fue remuestreada antes de la compresión JPEG, ya que la periodicidad introducida por el remuestreo oculta el de remuestreo.

Babak Mahdian y Stanislav Saic (Academy of Sciences of the Czech Republic, Praga, República Checa, 2008) [26] se basaron en algunos pasos principales para detectar el remuestreo: selección de ROI (Region Of Interest), cálculo de las derivadas de un orden alto (dos o superior), la transformada de Radón (hace posible la detección de la rotación) y la búsqueda de la periodicidad por medio de la DFT. Este método es evaluado para imágenes con formato TIFF y JPEG con factores de calidad de 95, 97 y 100. También adicionaron ruido blanco con un factor de 20, 30, 40, y 50. Primero el método es aplicado a las filas y después a las columnas de la imagen investigada. También intentan encontrar el factor de escalado usando la posición de los picos. Esto se llevó a cabo para factores de escala mayores que 1,03. El proceso de estimación se basó en la búsqueda del máximo global en la salida del método. La precisión de detección fue cercana al 100 %. Lo mismo se llevó a cabo para imágenes de formato TIFF con ángulos de rotación superiores a 3. La precisión de detección estaba nuevamente cerca del 100 %. Otra ventaja del método propuesto, en comparación con Popescu y Farid, es que no necesita ningún parámetro de inicialización que pueda afectar en gran medida los resultados obtenidos.

Babak Mahdian y Stanislav Saic (Institute of Information Theory and Automation of the ASCR, Czech Republic, 2008) [27] presentaron un método ciego capaz de encontrar rastros de remuestreo e interpolación. El método propuesto, así como otros detectores de interpolación / remuestreo existentes, es muy sensible al ruido. Por lo tanto, también proponen un método simple capaz de dividir una imagen investigada en varias particiones con niveles de ruido homogéneos. Agregar ruido aleatorio localmente puede causar inconsistencias en el ruido de la imagen. Por lo tanto, la detección

de varios niveles de ruido en una imagen puede significar manipulación. El método de detección de interpolación se basa en unos pocos pasos principales: selección de la región de interés (ROI), cálculo de la derivada de segundo orden de la señal, transformada de radón y búsqueda de la periodicidad mediante la DFT. El método es aplicado a las columnas. El análisis de las inconsistencias de ruido consiste de: Transformada wavelet, sub-banda de mosaico con bloques no superpuestos, estimación de la varianza del ruido para cada bloque y fusión de bloques. El método propuesto no puede encontrar las regiones dañadas, cuando la degradación del ruido es muy pequeña.

Matthias Kirchner (Institute for System Architecture, Technische Universität Dresden, Germany, 2008) [28] estudió el trabajo de Popescu y Farid, este no proporciona una relación explícita sobre cómo una transformación particular influirá en la salida del detector en el dominio espacial y de frecuencia, Matthias Kirchner presentó una versión acelerada del detector original de Popescu y Farid que elude los elementos más exigentes desde el punto de vista computacional en el proceso de detección y al mismo tiempo proporciona una detección igualmente confiable del remuestreo. Las modificaciones al algoritmo son dos: 1) reemplazar la estimación de algoritmo EM de los pesos escalares α por un filtrado lineal con coeficientesα preestablecidos, 2) procedimiento para detectar automáticamente la presencia de picos característicos en el espectro de un mapa de probabilidad.

Ye Zhu, Xuanjing Shen y Haipeng Chen (College of Computer Science and Technology, Jilin University, China, 2016) [29] proponen un método para la identificación de imágenes alteradas por copy-move que pueden tener transformaciones geométricas como la rotación y el escaldo, estas transformaciones geométricas se reducen a un remuestreo. El algoritmo tiene los siguientes pasos: primero, establecen un espacio de escala gaussiano; segundo, extraen los puntos claves mediante Orientated FAST and Rotated BRIEF (ORB) en cada espacio de escala; tercero, revierten las coordenadas de los puntos clave orientados a la imagen original y hacen coincidir las características ORB entre cada dos puntos clave diferentes utilizando la distancia de Hamming; finalmente, eliminan los puntos clave coincidentes falsos utilizando el algoritmo RANSAC. Los resultados son presentados en curvas ROC (Receiver Operating Characteristic) donde se aprecia que el método propuesto es efectivo respecto a las transformaciones geométricas, como la rotación y el escaldo, y es robusta cuando la imagen esta distorsionada por desenfoque gaussiano, ruido blanco gaussiano y compresión JPEG.

Gul Muzaffer, Ozge Makul, Beste Ustubioglu y Guzin Ulutas (Departmant of Computer Engineering Karadeniz Technical University Trabzon, Turkey, 2016) [30] estos autores desarrollaron una metodología, invariante a la escala y a la rotación, para detectar una región copiada por medio de la extracción de puntos claves usando ORB (Orientated FAST and Rotated BRIEF), además,

usan filtros Gabor. Primero, se extrae la información de textura de la imagen con filtros Gabor y luego ecualizan los histogramas; segundo, detectan los puntos clave y descriptores ORB de la imagen filtrada; tercero, hacen la coincidencia de puntos claves por medio de la distancia de Hamming para detectar la zona falsificada; Finalmente remueven las falsas coincidencias por medio del algoritmo RANSAC.

Capítulo 4

Marco Teórico

El uso y valor de la información digital ha ido en aumento en los últimos años, es por esto, que cuando se realiza un crimen la información puede estar guardada de forma digital. La informática forense aparece como un apoyo a la justicia, procura descubrir e interpretar la información en los medios electrónicos para establecer los hechos. En la Fig. 4.1 se muestran las divisiones de la informática forense, la forensia de redes se encarga de hacer el seguimiento de un evento en la red a través del rastro que deja en la infraestructura de datos, la computación forense analiza la información almacenada en un equipo de computo y la forensia digital aplica los mismos procedimientos de la criminalística tradicional a los medios informáticos para apoyar los procesos jurídicos [31].

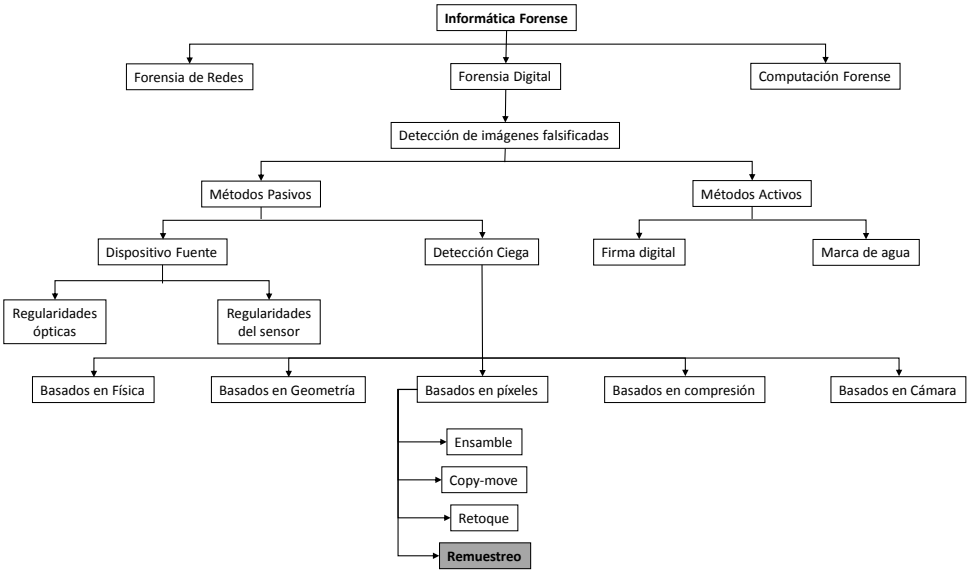


Figura 4.1: Informática forense y técnicas de falsificación de imágenes.

Dentro de la forensia digital se encuentra la detección de imágenes falsificadas, la cual tiene

dos enfoques principales que son la autenticación pasiva y la activa. Esta segunda se trata de marcas de agua o firmas digitales, las manipulaciones introducen información en la imagen para luego verificar si la imagen ha sido alterada o no. Los enfoques activos requieren acceso al proceso de captura o personal autorizado para incrustar la información, en caso de no conocer información de la firma digital o marca de agua es imposible usar este tipo de técnicas, lo cual es una desventaja. Los métodos de detección pasivos se enfocan en la información que se tiene de la imagen, no necesitan información adicional, se basa en buscar los rastros dejados por la manipulación en la propia imagen. En la Fig. 4.1 se muestran las divisiones y subdivisiones de los métodos pasivos. Los métodos basados en el dispositivo fuente buscan características que deja la cámara, para ello se requiere información del dispositivo pero en un proceso puede que no se cuente con esta información adicional. Las técnicas que no requieren información adicional a la imagen misma son llamadas técnicas ciegas. Estas se dividen en cinco aspectos: la física que busca inconsistencias de la luz, la geometría busca las posiciones relativas de los objetos respecto a la cámara, la compresión o formato se centran en características de artefactos de bloqueo o utilizan el factor de calidad para detectar la falsificación, la cámara se utiliza para identificar características de los dispositivos que generan las imágenes y los basados en píxeles que trabajan alrededor de las irregularidades estadísticas generadas por la falsificación [32, 33].

Las técnicas basadas en píxeles se dividen en cuatro, dependiendo del tipo de falsificación, estas son: copiado, ensamble, remuestreo y retoque. El copiado consiste en copiar y pegar secciones de la imagen en ella misma, el ensamble es la composición de de varias imágenes para crear una sola, el retoque quita una o varias porciones de la imagen y se usan algoritmos para rellenar estos espacios [34, 35]. La siguiente sección se dedicará a la falsificación por remuestreo.

4.1. FALSIFICACIÓN POR REMUESTREO

El remuestreo consiste es copiar y pegar una porción de la imagen en ella misma adicionando transformaciones geométricas como el escalado y la rotación. Esto se hace para crear consistencia visual en una imagen falsificada. Este tipo de transformaciones requieren un remuestreo de la porción de imagen, cambiando la estadística de la misma [1, 36]. Las transformaciones geométricas son un filtrado que implica crear nuevos píxeles a partir de la interpolación, lo que introduce una correlación entre las muestras de la imagen. Transformar una imagen discreta en otra introduce perdida significativa en la calidad de la imagen por ello se usa diferentes tipos de interpolación como: nearsest-neighbor, bilineal, bicúbica, catmull-rom, cubica B-spline, Mitchell-netravali, lanczos-2 y lanczos-3 [37]. El remuestreo de una señal es un proceso lineal, considerando una señal de m muestras, el número de estas puede ser incrementado o disminuido en un factor p/q a n muestras

en tres pasos:

1. up-sample: crea una nueva señal $x_u[pt]$con pm muestras, donde $x_u[pt] = x[t]$, $t = 1, 2, 3, ..., m$, y $x_u[t] = 0$ en otro caso.

2. interpolación: se hace una convolución de la imagen con un filtro pasa bajo, $x_i[t] = x_u[t] \star h[t]$.

3. down-sample: crea una nueva señal $x_d[t]$con n muestras, donde $x_d[t] = x_i[qt]$, $t = 1, 2, 3, ..., n$. la señal remuestreada es $x_d[t]$.

La operación de remuestreo varia en el segundo paso dependiendo del tipo de interpolación (filtro) que se use, los mas comunes son la interpolación bicúbica y bilineal. En forma vectorial el proceso anterior es descrito por la siguiente ecuación:

$$x_d[t] = A_{p/q} x[t]$$

Donde la matriz A de tamaño n x m representa los tres pasos del remuestreo, una matriz de factor de *4/3* tiene la siguiente forma:

$$A_{4/3} = \begin{bmatrix} 1 & 0 & 0 & 0 & \\ 0{,}25 & 0{,}75 & 0 & 0 & \\ 0 & 0{,}5 & 0{,}5 & 0 & \\ 0 & 0 & 0{,}75 & 0{,}25 & \\ 0 & 0 & 0 & 1 & \\ & & & & \ddots \end{bmatrix} \tag{4.1}$$

Como se puede observar en la ecuación 4.1, la matriz A introduce una correlación periódica entre los píxeles de la imagen. Cada píxel p es una combinación lineal de los p vecinos adyacentes [1, 36]. En una imagen real pueden ocurrir correlaciones entre los píxeles pero es improbable que sean periódicas. Para detectar una imagen alterada por remuestreo lo que se hace es buscar correlaciones periódicas entre los píxeles de un bloque de la imagen, siendo posible encontrar un conjunto de muestras periódicas que se correlacionan de la misma manera con sus vecinos.

4.2. ORB - ORIENTATED FAST AND ROTATED BRIEF

Ethan Rubble, Vicent Rabaud, Kurt Konolige y Gary Bradski (Willow Garage, Menlo Park, California, 2011) [38] desarrollaron una alternativa para las tecnicas SIFT (Scale-Invariant Feature Transform) y SURF (Speeded-Up Robust Features) llamada ORB (Orientated FAST and Rotated

BRIEF), tiene una complejidad de computación baja. ORB puede resistir la transformación geométrica, como la escala y la rotación, y el procesamiento posterior, como el desenfoque, el ruido y la compresión JPEG. ORB tiene licencia libre para su uso y su implementación se encuentra en OpenCV. ORB esta compuesto por el detector FAST (Features from Accelerated Segment Test) y descriptor BRIEF (Binary Robust Independent Elementary Features). Tanto el detector como el descriptor fueron modificados para hacer a ORB invariante a la rotación, del tal modo que desarrollaron oFAST (FAST keypoint orientation) y rBRIEF (Rotation-Aware BRIEF).

- oFAST: FAST es un método eficiente para encontrar puntos claves. Los autores usan un filtro de de Harris corner para rechazar bordes y proporcionar un puntaje a cada punto clave. FAST no incluye un operador de orientación para ello se usa un operador de centroide el cual da un solo resultado. FAST toma un umbral para la intensidad entre el píxel del centro de una región circular y los píxeles externos, para ORB se uso un radio de nueve píxeles. Se usa una medida de Harris corner para ordenar los puntos claves, para N puntos claves se establece un umbral lo suficientemente bajo para obtener más de N puntos claves luego se ordenan acorde a la medida de Harris y se escogen los N puntos superiores. El esquema de FAST es complementado con una escala piramidal, ya que por si solo no produce características multiescala, en cada nivel se extraen puntos claves FAST filtrados por Harris. Para medir la orientación de las esquinas usan la intensidad del centroide, se asume que la intensidad de la esquina esta desplazada de su centro y esto se usa para asignar una orientación a dicha esquina. El momento de una porción de imagen se define en la Ec. 4.2:

$$m_{pq} = \sum x^p y^q I(, x, y) \tag{4.2}$$

  A partir de estos momentos se puede hallar el centroide y la orientación de la porción de imagen como se muestra en la Ec. 4.3:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

$$\theta = arctan^2(m_{01}, m_{10}) \tag{4.3}$$

  Para el cálculo de la orientación se usa un parche de imagen circular de radio r de tal forma que x y y recorran valores desde $-r$ a r. Se construye un vector desde el centro de la esquina O hasta el centroide C, $\overrightarrow{OC}$.

- rBRIEF: BRIEF es un descriptor de características que usa pruebas binarias entre los píxeles en una región de imagen suavizada, es muy sensible a la rotación en un plano. BRIEF se desarrollo a partir de las pruebas binarias que se usa para entrenar un árbol de decisiones,

una vez entrenado un conjunto de 500 puntos claves o menos los arboles pueden devolver una etiqueta para cualquier punto clave. Los autores buscaron la prueba menos sensible a la orientación. Rubble usa una distribución gaussiana al rededor de la porción de imagen con un vector de $n = 256$. La imagen es suavizada y cada punto de prueba es una subventana de 5×5 de una porción de 31×31 píxeles. BRIEF falla cuando la rotación en el plano es de unos pocos grados, para ello Rubble sugiere dirigir a BRIEF de acuerdo a la orientación de los puntos claves. Para cualquier conjunto de n características binarias, con localización (x_i, y_i), se construye la matriz S,

$$S = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ y_1 & x_2 & \dots & y_n \end{pmatrix}$$

Usando la orientación θ del parche descrita en la ecuación 4.3 se construye $S_\theta = R_\theta S$, una versión dirigida de S, con la respectiva matriz de rotación R_θ. Así se obtiene un operador dirigido para BRIEF, definido de la siguiente manera:

$$g_n(p, \theta) = f_n(p) \mid (x_i, y_i)$$

Se discretiza el ángulo en incrementos de doce grados y se hace una búsqueda de patrones BRIEF. Mientras la orientación θ del punto clave sea consistente en todas las vistas, se utilizará el conjunto correcto de puntos S_θ para calcular su descriptor. Una de las propiedades de BRIEF es que cada característica de bit tiene una gran variación y una media cercana a 0,5, una media de este valor da una varianza para cada característica de 0.25. Una vez que BRIEF se orienta a lo largo de la dirección del punto clave, la media se desplaza a un patrón más distribuido (los puntos clave de las esquinas orientadas presentan un aspecto más uniforme a las pruebas binarias). La alta varianza hace que una característica sea más discriminativa, ya que responde de manera diferente a las entradas. Para el problema de la pérdida de variación en BRIEF dirigido y para reducir la correlación entre las pruebas binarias, se desarrolla un método de aprendizaje para elegir un subconjunto de pruebas binarias, en el cual, se busca entre todas las pruebas binarias posibles encontrar aquellas que tengan una varianza alta (y media cercana a 0.5), además de no estar correlacionadas.

ORB entrega como máximo 500 puntos claves con orientación obtenidos a través oFAST , a cada punto clave le corresponde un vector de 256 características binarias extraídas por rBRIEF.

4.3. P-MAP - MAPAS DE PROBABILIDAD

En el 2005 Popescu y Farid [1], proponen una técnica la cual usa mapas de probabilidad (p-map) para detectar la falsificación por remuestreo en una imagen. Basados en la periodicidad de la matriz de remuestreo 4.1 se puede observar que ciertas muestras tiene la misma correlación entre sus vecinos. La forma de la correlación se puede encontrar mediante el tamaño del vecindario, N, y un conjunto de pesos $\overrightarrow{\alpha}$ que satisfagan las siguiente ecuación:

$$\overrightarrow{a_i} = \sum_{k=-N}^{N} \alpha_k \overrightarrow{a}_{i+k}$$

Donde $\overrightarrow{a_i}$ es la i-ésima fila de la matriz remuestreo. Si se conoce la forma de las correlación, α, se podría determinar que píxeles satisfacen la Ec. 4.4.

$$y_i = \sum_{k=-N}^{N} \alpha_k y_{i+k} \tag{4.4}$$

En la vida real los píxeles no están correlacionados de una forma especifica y conocida. Para determinar si una imagen ha sido remuestreada se hace uso del algoritmo EM (Expectation/Maximitation), que consiste en dos pasos iterativos, se definen dos modelos, $M1$ el píxel esta correlacionado con sus vecinos y $M2$ el píxel no esta correlacionado con sus vecinos. En el paso E se calcula la probabilidad de que el píxel pertenezca al modelo $M1$ o $M2$, la probabilidad de que cada muestra pertenezca al modelo M1 puede obtenerse usando la regla de Bayes. En el paso M se busca la correlación especifica que existe entre los píxeles. En el paso E la probabilidad de que la muestra y_i pertenezca al modelo $M1$ se calcula de la siguiente manera:

$$Pr\{y_i \mid y_i \in M1\} = \frac{1}{\sigma\sqrt{2\pi}}\, exp\left[\frac{-\left(y_i - \sum_{k=-N}^{N} \alpha_k y_{i+k}\right)^2}{2\sigma^2}\right] \tag{4.5}$$

Donde la varianza, σ, es calculada en el paso M. El paso E requiere conocer el valor de $\overrightarrow{\alpha}$, inicialmente se escoge aleatoriamente y en el paso M se recalcula buscando disminuir el error cuadrático de la siguiente expresión.

$$E(\overrightarrow{\alpha}) = \sum_i \omega(i)\left(y_i - \sum_{k=-N}^{N} \alpha_k y_{i+k}\right)^2 \tag{4.6}$$

Adicional se calcula el gradiente respecto a $\overrightarrow{\alpha}$, para ayudar a disminuir la función de error, igua-

lando a cero y resolviendo para $\overrightarrow{\alpha}$. Los pasos E y M son iterativos hasta lograr un valor aceptable para $\overrightarrow{\alpha}$. Típicamente el algoritmo converge a 4.7:

$$\overrightarrow{\alpha} = \begin{bmatrix} -0{,}25 & 0{,}5 & -0{,}25 \\ 0{,}5 & 0 & 0{,}5 \\ -0{,}25 & 0{,}5 & -0{,}25 \end{bmatrix} \tag{4.7}$$

Una vez se obtiene el mapa de probabilidad de la imagen, se hace la transformada de Fourier, si la imagen tiene remuestreo en su transformada se observa un patrón periódico.

4.4. ESPECTRO DE ENERGÍA

El espectro de energía es una representación de la distribución de la energía en función de la frecuencia. La derivada de segundo orden en una imagen es un filtro paso altos que cumple con la función de destacar los detalles finos de la imagen, un ejemplo para este tipo de filtro es el Laplaciano 4.8. Como se ha mencionado anteriormente la detección de remuestreo busca la correlación entre píxeles o patrones periódicos para ello se recurre a la transformada de Fourier la cual pasa la imagen del dominio espacial al dominio de la frecuencia donde son mas evidentes los patrones periódicos introducidos por el remuestreo en la imagen.

$$h = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \tag{4.8}$$

4.5. MARKOV

En el campo de la probabilidad el modelo de Markov es un proceso en el que la probabilidad de que ocurra un evento esta dada por el evento inmediatamente anterior. En [39] los autores proponen extraer características de Markov en el dominio especial y DCT para detectar imágenes falsificadas con splaicing (empalme). En la detección de imágenes alteradas se buscan bordes, para ello se calculan derivadas direccionales en dirección horizontal, vertical, diagonal mayor y diagonal menor, de acuerdo a las siguientes ecuaciones:

$$E_v = I(u,v) - I(u+1,v) \qquad 1 \le u \le S_u - 1,\ 1 \le v \le S_v$$

$$E_h = I(u,v) - I(u,v+1) \qquad 1 \le u \le S_u,\ 1 \le v \le S_v - 1$$

$$E_d = I(u,v) - I(u+1,v+1) \qquad 1 \leq u \leq S_u - 1,\ 1 \leq v \leq S_v - 1$$

$$E_m = I(u+1,v) - I(u,v+1) \qquad 1 \leq u \leq S_u - 1,\ 1 \leq v \leq S_v - 1$$

Donde I es la imagen de entrada y $S_u \times S_v$ su dimensión, las derivadas direccionales dan como resultado una imagen de las mismas dimensiones de I. Para reducir la dimencionalidad de la matriz de transición de probabilidad (TPM) se usa un umbral T de tal forma que todos los valores de las derivadas direccionales se encuentran entre $-T$ y $+T$ con solo $(2T+1)$ valores posibles.

Para obtener la matriz de transición de probabilidad se usa un proceso aleatorio de Markov, el cual describe la correlación entre píxeles. Este proceso se realiza sobre las matrices umbralizadas, TPM tendrá dimensiones de $(2T+1)\times(2T+1)$ para cada dirección, entonces para una imagen en el dominio espacial o DCT se tendrán $4\times(2T+1)\times(2T+1)$ características de Markov.

Las matrices de probabilidad de transición de una imagen se calcula en dirección horizontal, vertical, diagonal mayor y diagonal menor, para ello se usan las ecuaciones 4.9, 4.10, 4.11 y 4.12.

$$P[T_h(u+1,v) = j \mid T_h(u,v) = i] = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(T_h(u,v)=i,\, T_h(u+1,v)=j)}{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v}\delta(T_h(u,v)=i)} \tag{4.9}$$

$$P[T_v(u,v+1) = j \mid T_v(u,v) = i] = \frac{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(T_v(u,v)=i,\, T_v(u,v+1)=j)}{\sum_{u=1}^{S_u}\sum_{v=1}^{S_v-2}\delta(T_v(u,v)=i)} \tag{4.10}$$

$$P[T_d(u+1,v+1) = j \mid T_d(u,v) = i] = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v-2}\delta(T_d(u,v)=i,\, T_d(u+1,v+1)=j)}{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v-2}\delta(T_d(u,v)=i)} \tag{4.11}$$

$$P[T_h(u,+1v) = j \mid T_h(u,v) = i] = \frac{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_v-2}\delta(T_m(u+1,v)=i,\, T_m(u,v+1)=j)}{\sum_{u=1}^{S_u-2}\sum_{v=1}^{S_{v-2}}\delta(T_m(u+1,v)=i)} \tag{4.12}$$

Donde $\delta(A=i,B=j) = \begin{cases} 1 & A=1,\ B=j \\ 0 & en\ otro\ caso \end{cases}$

El resultado de estas ecuaciones forman la matriz de transición de probabilidad.

Capítulo 5

Desarrollo metodológico

Los algoritmos implementados para la detección de falsificación del tipo remuestreo se elaboraron a partir de trabajos publicados en revistas indexadas. Estos algoritmos se ejecutaron mediante el software MATLAB 2017b, en un equipo Dell de 64bits, 8GB de RAM con Windows 10 y un procesador Intel® Xeon.

5.1. IMPLEMENTACIÓN DE LAS TÉCNICAS

Durante la revisión del estado del arte se escogieron cuatro técnicas, las cuales pudieran ser implementadas en las condiciones del laboratorio Computer Vision and Maching Learning Laboratory. La clasificación de imágenes en originales y falsificadas por remuestreo es un problema de reconocimiento de dos clases, en la implementación de las siguientes técnicas se asumió una clasificación binaria, 0 y 1, donde 0 es una imagen original y 1 una imagen con remuestreo.

5.1.1. Técnica de ORB (Orientated FAST and Rotated BRIEF)

ORB es un extractor de puntos claves con sus respectivas características. Su implementación se hizo en base al articulo de Irene Amerini [40] cambiando a SIFT por ORB y modificando los umbrales.

1. Primero se convirtió la imagen en escala de grises y se extrajeron los puntos claves por medio de ORB. Dada una imagen I se obtiene un vector $S = [s_1, s_2, \ldots, s_n]$ el cual almacena la posición de los n puntos claves que se hallaron, de tal forma que $s_i = [x_i, y_i]$, donde x e y representan la ubicación en filas y columnas del punto clave. Por ultimo se crea un vector $F = [f_1, f_2, \ldots, f_n]$ donde $f_i$es un vector de 32 características correspondiente al i-ésimo punto clave.

2. Coincidencia de características por medio de g2NN. Una imagen falsificada por remuestreo implica que halla una región igual a otra pero con transformaciones geométricas, ORB al ser invariante a la escala y la rotación va a generar vectores de características idealmente iguales para ambas regiones. Para detectar que dos puntos claves tienen las mismas características se usa una distancia euclidiana con los dos vecinos mas cercanos (2NN). Este método falla cuando se ha falsificado múltiples ocasiones la misma región para ello se calculó la distancia euclidiana entre todos los vectores de características de los puntos claves. Teniendo en cuenta la relación $\frac{d_i}{d_{i+1}} \leq \tau$, donde d es la distancia entre pares y τ un umbral definido en $0,5$, se genera un grupo de pares coincidentes que se almacenan en un vector $P = [p_1, p_2, \ldots, p_m]$.

3. Agrupamiento J-Linkage de puntos claves. Se usó un agrupamiento HAC (Hierarchical Agglomerative Clustering) este método tiene en cuenta las coordenadas de lo pares coincidentes. Presenta problemas cuando las zonas coincidentes están muy cercanas. Para solucionar este inconveniente los autores proponen una técnica de agrupamiento que trabaja en el dominio de la transformación (J-linkage). La agrupación se hizo con una selección aleatoria de los pares p_i coincidentes para generar transformaciones afines. Para cada par, se define un vector de conjunto de preferencias PS que indica que transformaciones prefiere el par.

La implementación del algoritmo de J-linkage fue realizada por la ingeniera fisica Maria Caminila Maya Piedrahita.

5.1.2. Técnica de Mapas de Probabilidad

Basado en el 2016 Nguyen [14], se implementó los mapas de probabilidad para detectar el remuestreo. Se uso la convergencia del paso M para la matriz de pesos expresada en la ecuación 4.7 del algoritmo EM anunciado por Popescu y Farid en el 2005, de esta forma se disminuye el costo computacional ya que deja de ser un algoritmo iterativo. los pasos que se siguieron fueron los siguientes:

1. Primero se convirtió la imagen a escala de grises y se calculo el mapa de probabilidad (p-map) por medio de la ecuación 4.5con valores de $\overrightarrow{\alpha}$ definidos en 4.7 para un tamaño de vecindario de $N = 3$, Los autores lo llaman pseudo p-map, la varianza que se uso fue de $\sigma = 0{,}075$ y $P_0 = 1/255$ (255 ya que es el máximo valor que alcanzan los píxeles de la imagen al ser escala de grises). El mapa de probabilidad se forma a partir de cada píxel por medio de la ecuación $w_i = \frac{Pi}{Pi+P_0}$.

2. Se aplicó la transformada de Radon discreta al pseudo p-map, para ellos se uso la función de MatLab Radon. El conjunto de ángulos fue de 0° a 179° en pasos de 1°. Después de la transformada de Radon se obtiene un conjunto de vectores proyectados organizados en una

matriz. Para evidenciar patrones periódicos fuertes (picos) se aplico la transformada de Fourier a los vectores proyectados. Debido a la periodicidad de los pseudo p-map, se mostrarán picos significativos en el caso de las imágenes remuestreadas.

3. Se extrajo como características el resultado de la transformada de Fourier a las proyecciones de Radon del pseudo p-map, el cual es un vector de 180 características que se usaron para alimentar una máquina de soporte vectorial (SVM) y realizar la clasificación de imágenes. Se usó la aplicación de MatLab Classification Learner para entrenar y evaluar una SVM cuadrática.

5.1.3. Técnica del Espectro de energía

En el 2012 los autores Qian et al. [19] proponen usar el espectro de energía para detectar el remuestreo mediante lo que ellos llaman RTRSD (rotation tolerant resampling detection). Se convirtió la imagen en escala de grises, se calculo la energía del espectro con RTRSD y se entrenó una máquina de soporte vectorial con kernel cuadrático. RTRSD consiste en encontrar la derivada de segundo orden a la imagen I, para lo cual se realizó un filtro Laplaciano 4.8 a la imagen usando la función imfilter de MatLab, se seleccionó una constante $k = \frac{1}{3}N$ donde N es el número de filas, se calculó el espectro de energía con la FFT a cada $i \times \frac{N}{k}$ fila, con $i = 1, 2, 3, ..., k$ (a cada tres filas se le calculó la FFT) de esta forma se obtuvo un grupo de vectores $\alpha_1, \alpha_2, ..., \alpha_k$. El resultado de RTRSD es un vector α que contiene la suma del espectro de energía de cada fila, $\alpha = \sum_{i=1}^{k} \alpha_i$, este vector contiene las características de la imagen que se usaron para el clasificador. El tamaño del vector α es $1 \times M$, donde M es el numero de columnas de la imagen, ya que las imágenes usadas tienen diferente numero de filas se escalo el tamaño del vector α a $1x200$ características. Para esto se halló un factor $f = \frac{M}{200}$, se calculó la nueva posición $j = f \times n$ con $n = 1, 2, ..., 200$, de tal forma que se obtuvo un nuevo vector $RP(n) = \alpha(j)$ el cual se utilizó para el clasificador de máquinas de soporte vectorial con un kernel cuadrático.

5.1.4. Técnica de Markov

Al ser Markov un método que busca la probabilidad de eventos se puede utilizar para detectar remuestreo. El método consiste en la extracción de características, reducción de características y la detección de la falsificación por medio de SVM. Se extraen características del dominio especial y DCT que contiene cambios estadísticos o de la correlación de los píxeles obtenidos por Markov. Primero se convirtió la imagen a escala de grises y se dividió en bloques de 8×8 píxeles sin overlapping, se recorto la imagen para que su tamaño coincidiera con los bloques que cambian en ella, y se aplico DCT a cada bloque, los coeficientes DCT se aproximaron a valores absolutos y enteros, se calculó la matriz de probabilidad de transición (TPM) con el uso de las Ec. 4.9, 4.10,

4.11 y 4.12 para la imagen en el dominio DCT y en el dominio espacial, lo cual da un vector de 648 características. Se usó la aplicación de MatLab Classification Learner para la reducción de características y para entrenar y evaluar una SVM. La implementación para el cálculo de la matriz de probabilidad de transición fue realizada por el ingeniero electrónico Cristian David Lopez Robayo, estudiante de la maestría en instrumentación física, miembro del Grupo de Investigación en Robótica Aplica GIRA de la Universidad Tecnológica de Pereira.

5.2. EVALUACIÓN DE LAS TÉCNICAS

Las técnicas fueron evaluadas en tres bases datos las cuales se especifican en la sección 6.2.1. Para validar los resultados obtenidos con cada base de datos, se hizo uso de la Matriz de Confusión y la curva ROC (Receiver Operating Characteristic) las cuales se explican en las secciones siguientes.

5.2.1. Base de datos

Se utilizaron tres bases de datos encontradas en la literatura, DATASET [41], COVERAGE [42] y MICC2000 [43]. Las bases de datos están diseñadas para copy-move pero contienen imágenes con transformaciones geométricas, como la rotación y el escalado. Las imágenes o bases de datos utilizadas en los artículos relacionados con remuestreo son falsificadas por los mismos autores y no son de acceso libre. En el cuadro 5.1 se muestran las características de cada base de datos, como el formato, la cantidad de imágenes y el tamaño. Además, se especifica el número de imágenes originales y con remuestreo, ya que al ser bases de datos para copy-move algunas imágenes no contenían transformaciones geométricas por lo tanto no se tuvieron en cuenta en las pruebas.

Base de datos	**Formato**	**Total imágenes**	**Imágenes originales**	**Imágenes con remuestreo**	**Imágenes evaluadas**	**Tamaño**
DATASET	BMP	1040	20	920	940	768×1024
COVERAGE	TIF	200	100	100	200	368×431 a 768×1024
MICC2000	JPG	2000	650	1300	1950	2048×1536

Cuadro 5.1: Bases de datos.

5.2.2. Matriz de confusión

La matriz de confusión muestra el desempeño del algoritmo a través de una matriz 2x2 como se muestra en la Fig. 5.1. En este caso se tienen dos clases, cero y uno, un cero (0) representa una imagen original y un uno (1) representa una imagen falsificada. En la parte horizontal se tiene la clasificación correcta de las imágenes y en la vertical se tiene la clasificación que arrojo la técnica.

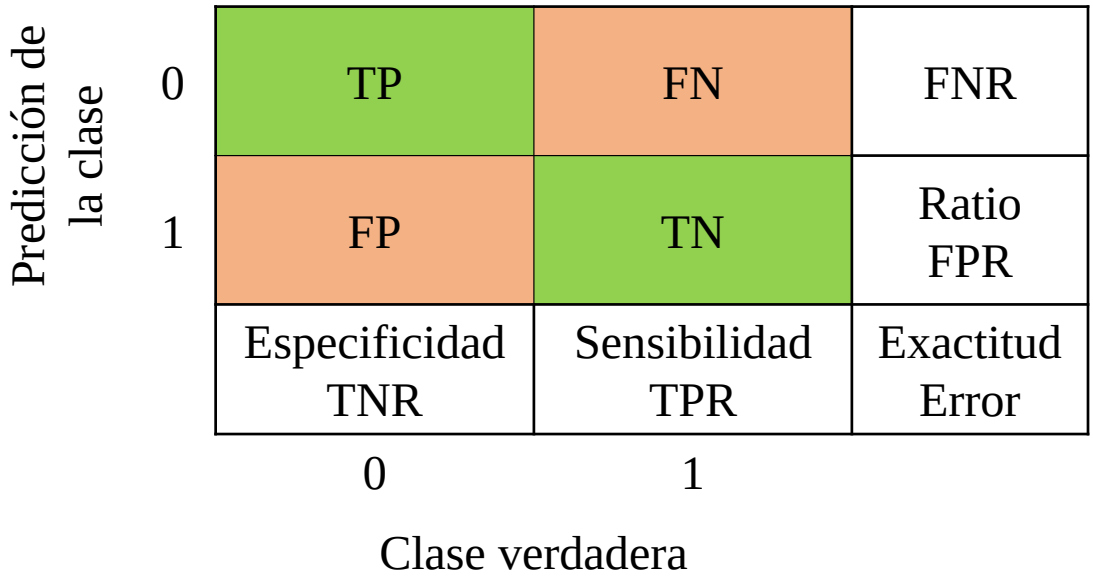


Figura 5.1: Matriz de confusión.

- TP (true positive) verdaderos positivos: Es la cantidad de muestras (imágenes) que el algoritmo clasifica como positivas (imágenes falsificadas) que realmente eran positivas (imágenes falsificadas).

- TN (true negative) verdaderos negativos: Es la cantidad de muestras (imágenes) que el algoritmo clasifica como negativas (imágenes originales) que realmente eran negativas (imágenes originales).

- FP (false positive) falsos positivos: Es la cantidad de muestras (imágenes) que el algoritmo clasifica como positivas (imágenes falsificadas) que realmente eran negativas (imágenes originales).

- FN (false negative) falsos negativo: Es la cantidad de muestras (imágenes) que el algoritmo clasifica como negativas (imágenes originales) que realmente eran positivas (imágenes falsificadas).

  De la matriz de confusión se obtienen las siguientes métricas: la **exactitud** (porción total de predicciones correctas o capacidad del algoritmo de predecir la clase real), el **error** (inexactitud o porción total de predicciones incorrectas), la **sensibilidad** (la capacidad del algoritmo para detectar imágenes falsas que realmente son falsas), la **especificidad** (La capacidad del algoritmo de detectar muestras imágenes originales que realmente son originales), **FPR** (tasa

de falsos positivos), **FNR** (tasa de falsos negativos), **la precisión** (es la capacidad del algoritmo de dar el mismo resultado en pruebas diferentes) y **F-measure** (medida de la precisión de una prueba). Las ecuaciones matemáticas para calcular estas variables son:

$$Exactitud \longrightarrow AC = \frac{TP+TN}{TP+TN+FP+FN} \tag{5.1}$$

$$Error = 1 - AC \tag{5.2}$$

$$Sensibilidad \longrightarrow TPR = \frac{TP}{TP+FN} \tag{5.3}$$

$$Especificidad \longrightarrow TNR = \frac{TN}{TN+FP} \tag{5.4}$$

$$FPR = \frac{FP}{FP+TN} \tag{5.5}$$

$$FNR = \frac{FN}{FN+TP} \tag{5.6}$$

$$Precisión = \frac{TP}{TP+FP} \tag{5.7}$$

Un buen algoritmo para la detección de imágenes falsificadas por remuestreo debe tener una exactitud, sensibilidad, especificidad y precisión cerca a uno o al 100 %. Por lo tanto el error, FPR y FNR debe ser cercano a cero.

5.2.3. Curva ROC (Receiver Operating Characteristic)

La curva ROC o Característica Operativa del Receptor es una gráfica que representa que tan optimo es el método utilizado, mediante el área bajo la curva (AUC - Area Under Curve) de los ejes coordenados, sensibilidad (vertical) y 1-especificidad (Horizontal). En la Fig. 5.2 se muestra la interpretación de la curva ROC. Un buen clasificador tiene un AUC mayor a 0,5, es decir su gráfica se encuentra en la región azul, por lo tanto el clasificador tendrá una alta sensibilidad y especificidad. Un mal clasificador tendrá un AUC menor o igual a 0,5, su gráfica se encontrará por debajo de la linea roja punteada y la sensibilidad y especificidad va ser baja.

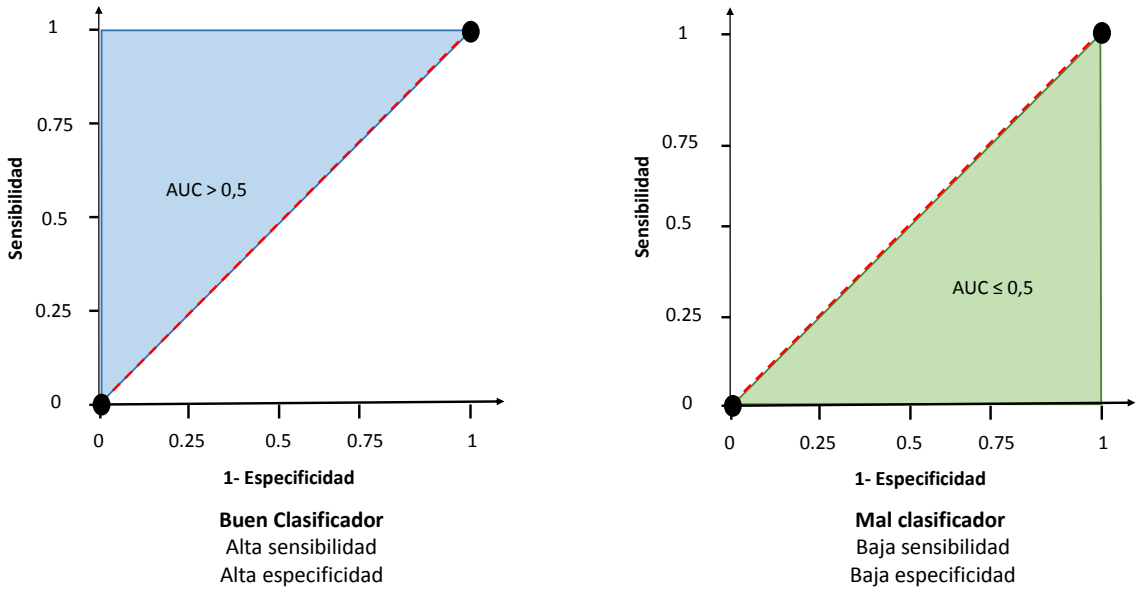


Figura 5.2: Curva ROC.

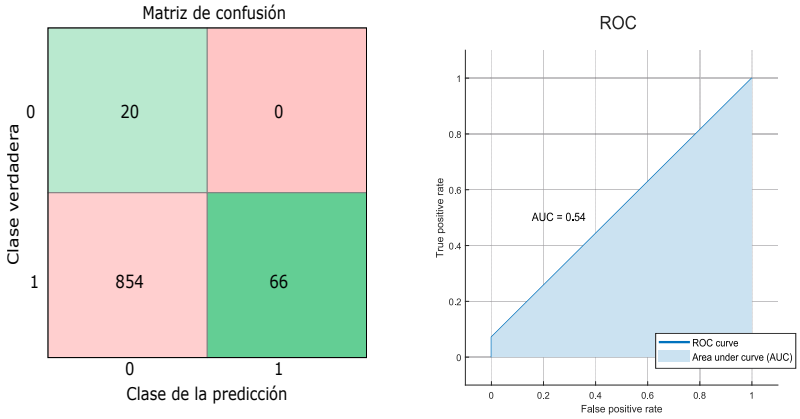
Capítulo 6

Resultados

En este capitulo se muestra la matriz de confusión y la curva ROC obtenidas en las bases de datos DATASET, COVERAGE y MICC2000 para las técnicas de ORB, mapas de probabilidad, espectro de energía y Markov.

6.1. RESULTADOS PARA LA TÉCNICA ORB

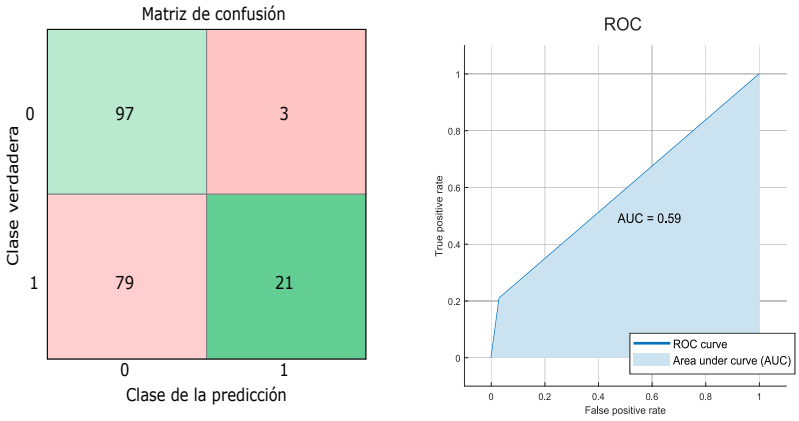
6.1.1. Base de datos DATASET



(a) Matriz de confusión (b) Curva ROC

Figura 6.1: Resultado de la técnica de ORB en la base de datos DATASET.

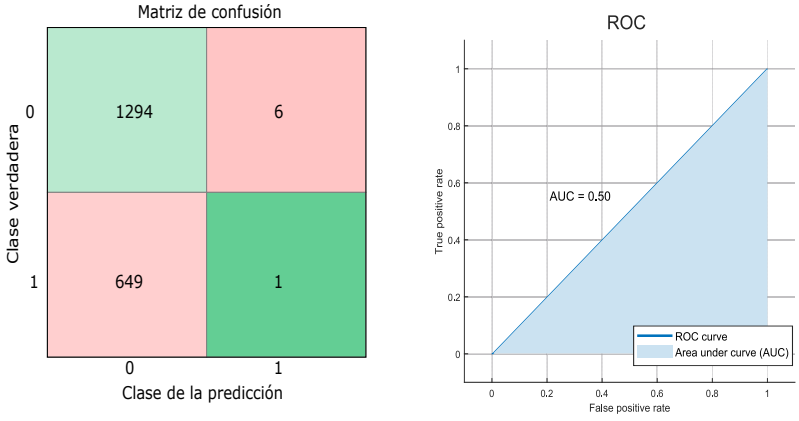
6.1.2. Base de datos COVERAGE



(a) Matriz de confusión (b) Curva ROC

Figura 6.2: Resultado de la técnica de ORB en la base de datos COVERAGE.

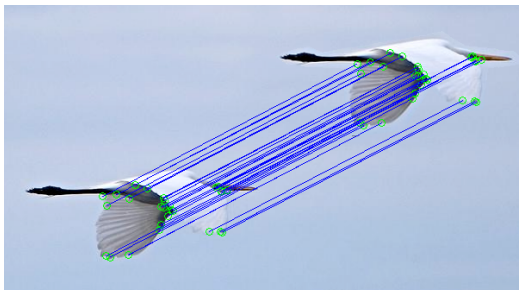
6.1.3. Base de datos MICC2000



(a) Matriz de confusión (b) Curva ROC

Figura 6.3: Resultado de la técnica de ORB en la base de datos MICC2000.

ORB es capaz de localizar la zona de la falsificación, en la Fig. 6.4 se aprecia los puntos claves y la coincidencia de puntos para las tres bases de datos.



(a) Coincidencia de los puntos claves en DATASET.

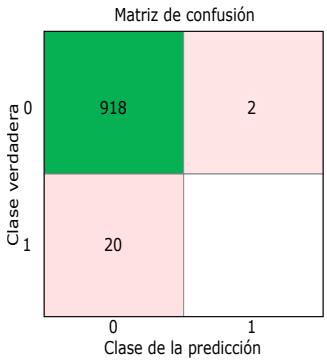
(b) Coincidencia de los puntos claves en COVERAGE.

(c) Coincidencia de los puntos claves en MICC2000.

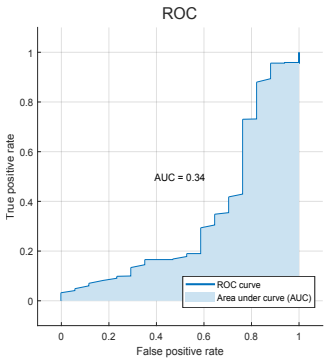
Figura 6.4: Coincidencia de puntos claves ORB en las tres bases de datos.

6.2. RESULTADOS PARA LA TÉCNICA DE MAPAS DE PROBABILIDAD

6.2.1. Base de datos DATASET



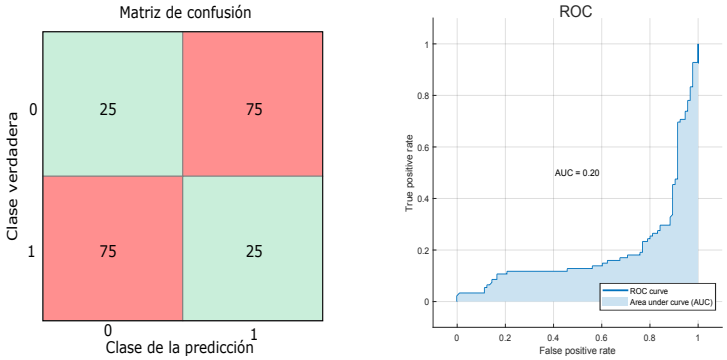
(a) Matriz de confusión



(b) Curva ROC

Figura 6.5: Resultado de la técnica de mapas de probabilidad en la base de datos DATASET.

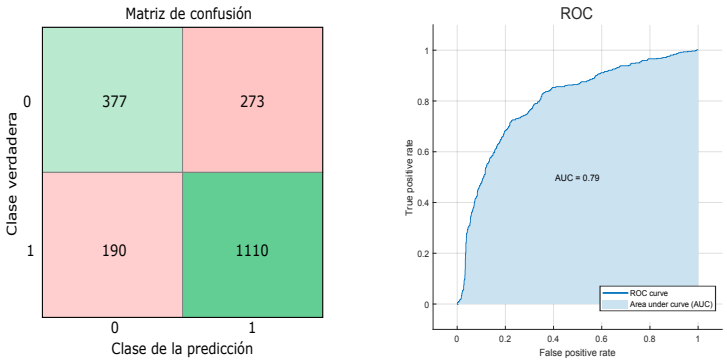
6.2.2. Base de datos COVERAGE



(a) Matriz de confusión (b) Curva ROC

Figura 6.6: Resultado de la técnica de mapas de probabilidad en la base de datos COVERAGE.

6.2.3. Base de datos MICC2000

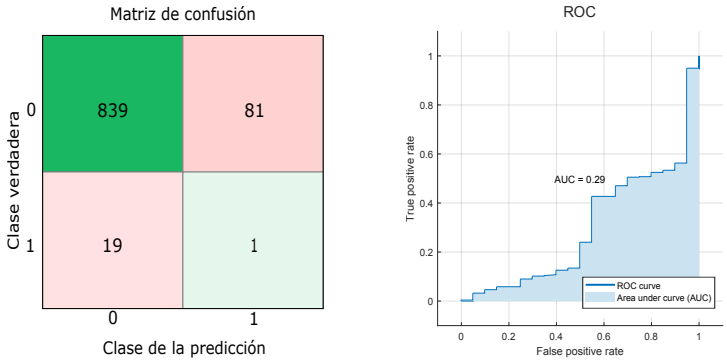


(a) Matriz de confusión (b) Curva ROC

Figura 6.7: Resultado de la técnica de mapas de probabilidad en la base de datos MICC2000.

6.3. RESULTADOS PARA LA TÉCNICA DEL ESPECTRO DE ENERGÍA

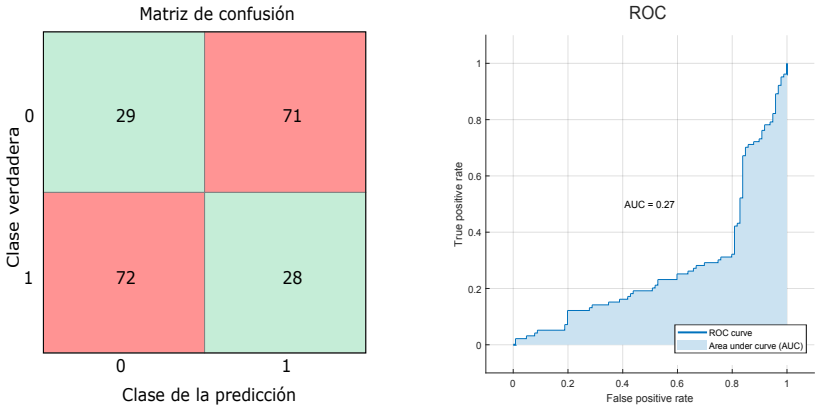
6.3.1. Base de datos DATASET



(a) Matriz de confusión (b) Curva ROC

Figura 6.8: Resultado de la técnica espectro de energía en la base de datos DATASET.

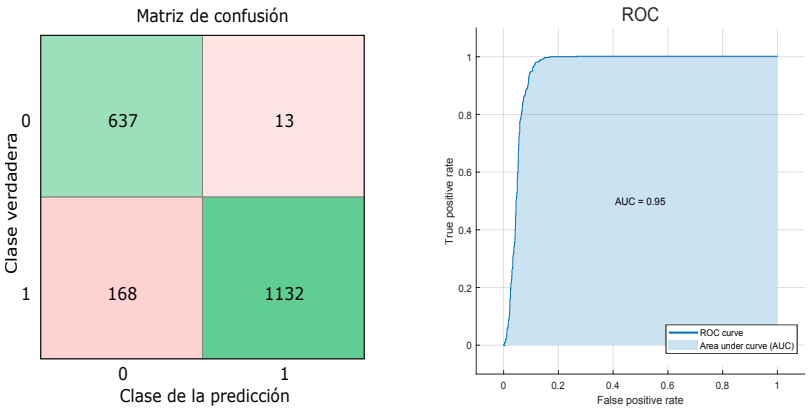
6.3.2. Base de datos COVERAGE



(a) Matriz de confusión (b) Curva ROC

Figura 6.9: Resultado de la técnica espectro de energía en la base de datos COVERAGE.

6.3.3. Base de datos MICC2000

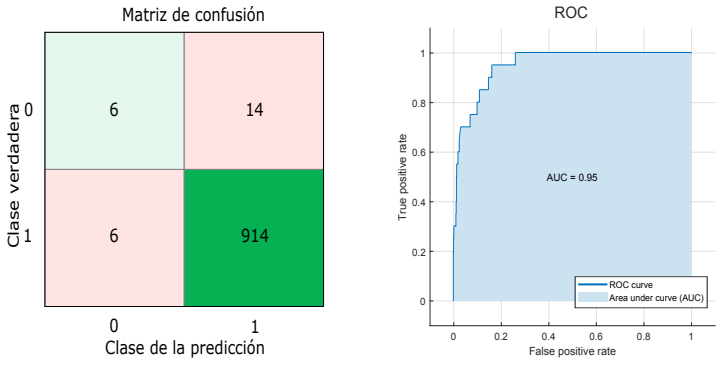


(a) Matriz de confusión (b) Curva ROC

Figura 6.10: Resultado de la técnica espectro de energía en la base de datos MICC2000.

6.4. RESULTADOS PARA LA TÉCNICA DE MARKOV

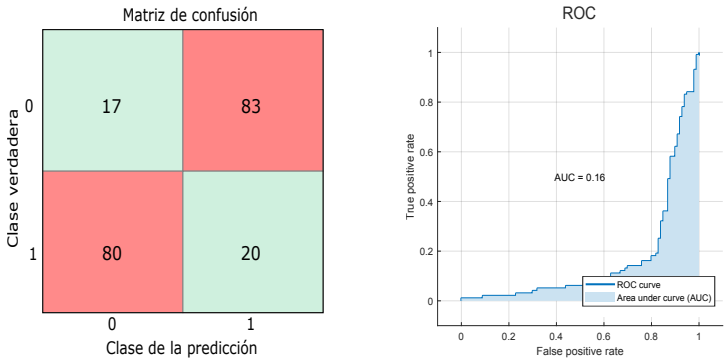
6.4.1. Base de datos DATASET



(a) Matriz de confusión (b) Curva ROC

Figura 6.11: Resultado de la técnica Markov en la base de datos DATASET.

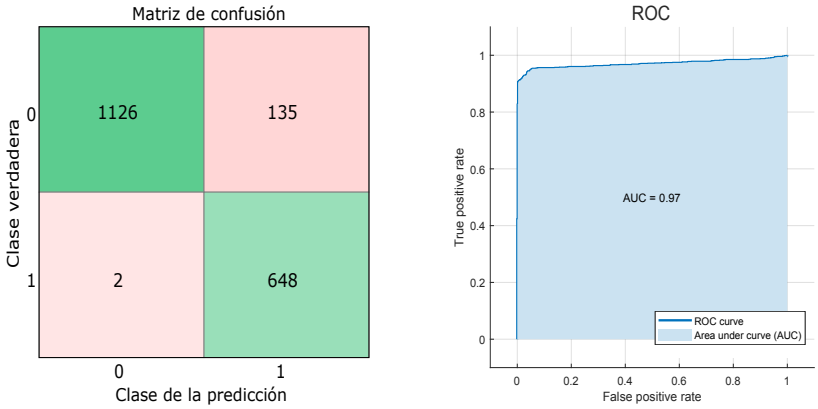
6.4.2. Base de datos COVERAGE



(a) Matriz de confusión (b) Curva ROC

Figura 6.12: Resultado de la técnica de Markov en la base de datos COVERAGE.

6.4.3. Base de datos MICC2000



(a) Matriz de confusión (b) Curva ROC

Figura 6.13: Resultado de la técnica de Markov en la base de datos MICC2000.

A partir de los resultados anteriores se obtuvieron las tablas 6.1, 6.2 y 6.3 donde se evidencia la exactitud, el error, la sensibilidad, la especificidad, la tasa falsos positivos, la tasa de falsos negativos, la precisión y el área bajo la curva de cada base de datos evaluada con cada técnica

DATASET	**Exactitud**	**Error**	**Sensibilidad**	**Especificidad**	**FPR**	**FNR**	**Precisión**	**AUC**
ORB	9,1 %	90,9 %	7,2 %	100,0 %	0,0 %	92,8 %	100,0 %	0,54
SP-map	97,7 %	2,3 %	0,0 %	99,8 %	0,2 %	100,0 %	0,0 %	0,34
Espectro de energía	89,4 %	10,6 %	5,0 %	91,2 %	8,8 %	95,0 %	1,2 %	0,29
Markov	97,7 %	2,3 %	99,3 %	30,0 %	70,0 %	0,7 %	98,5 %	0,95

Cuadro 6.1: Resultados de las cuatro técnicas en la base de datos DATASET.

COVERAGE	**Exactitud**	**Error**	**Sensibilidad**	**Especificidad**	**FPR**	**FNR**	**Precisión**	**AUC**
ORB	59,0 %	41,0 %	21,0 %	97,0 %	3,0 %	79,0 %	87,5 %	0.59
SP-map	25,0 %	75,0 %	25,0 %	25,0 %	75,0 %	75,0 %	25,0 %	0,2
Espectro de energía	28,5 %	71,5 %	28,0 %	29,0 %	71,0 %	72,0 %	28,8 %	0,27
Markov	18,5 %	81,5 %	20,0 %	17,0 %	83,0 %	80,0 %	19,4 %	0,16

Cuadro 6.2: Resultados de las cuatro técnicas en la base de datos COVERAGE.

MICC2000	**Exactitud**	**Error**	**Sensibilidad**	**Especificidad**	**FPR**	**FNR**	**Precisión**	**AUC**
ORB	66,4 %	33,6 %	0,2 %	99,5 %	0,5 %	99,8 %	14,3 %	0,50
SP-map	76,3 %	23,7 %	85,4 %	58,0 %	42,0 %	14,6 %	80,3 %	0,79
Espectro de energía	90,7 %	9,3 %	87,1 %	98,0 %	2,0 %	12,9 %	98,9 %	0,95
Markov	93,0 %	7,0 %	99,7 %	89,6 %	10,4 %	0,3 %	82,8 %	0,97

Cuadro 6.3: Resultados de las cuatro técnicas en la base de datos MICC2000.

Capítulo 7

Análisis de resultados y Discusión

Discusión

Markov, P-map y el espectro de energía obtuvieron el mayor valor para el área bajo la curva en la base de datos MICC2000 con un 0,97, 0,79 y 0,95 respectivamente y el menor en la base de datos COVERAGE con un 0,16, 0,2 y 0,27. Por el contrario, ORB obtuvo el mayor valor para el AUC en la base de datos COVERAGE con un 0,59 y el menor en la base de datos MICC2000 con un 0,50.

En la base de datos DATASET la técnica de mapas de probabilidad (P-map) y Markov obtuvieron la misma exactitud 97,7 % pero la sensibilidad de Markov fue mayor a la de P-map (99,3 % contra un 0 %) lo que indica que Markov tiene mejor capacidad que P-map para clasificar una imagen como falsa, que realmente es falsa. En el caso de la Especificidad ocurrió lo contrario; P-map obtuvo un 99,8 % contra el 30 % de Markov, así que P-map tiene una mejor capacidad que Markov para clasificar una imagen original, que realmente es original. ORB obtuvo la exactitud más baja en esta base de datos con un 9,1 %, aunque tuvo un 100 % de especificidad y 7,2 % en sensibilidad, lo que indica que a la mayoría de imágenes las clasifica como originales. En la base de datos COVERAGE se obtuvieron los valores más bajos para la exactitud de las técnicas P-map, espectro de energía y Markov con 25 %, 28,5 % y 18,5 % respectivamente. En esta base de datos la técnica que obtuvo la exactitud mayor fue ORB con un 59 %, las demás técnicas obtuvieron exactitud, especificidad y sensibilidad por debajo del 30 %. En la base de datos MICC2000 las técnicas de P-map, Espectro de energía y Markov obtuvieron porcentajes altos para la exactitud, sensibilidad y la especificidad al mismo tiempo. ORB logró obtener el mayor valor para la exactitud en esta base de datos, aunque, obtuvo un valor muy bajo para la sensibilidad lo que indica que este método no tiene la capacidad de clasificar imágenes falsas como falsas, que realmente son falsas en la base de datos MICC2000.

Las técnicas de Mapas de probabilidad y el Espectro de energía buscan picos en el dominio de

la frecuencia, estos picos pueden ser confundidos con los producidos por la compresión JPEG de la imagen ya que también es un proceso periódico y puede que no sean evidentes debido al ruido o al tamaño de de la región manipulada. Estos métodos también pueden ser sensibles cuando la imagen presenta bordes o cambios drásticos entre los píxeles.

Las técnicas de ORB y Markov no se encuentran implementadas en la literatura con el propósito de detectar imágenes alteradas por remuestreo, se decidió usar las debido a sus características. ORB es un detector de puntos claves invariante a la rotación y a la escala, debido a esto y con ayuda de la técnica propuesta por Amerini [43] para detectar copy-move se implementó dando un nuevo enfoque a la detección de remuestreo ya que normalmente las técnicas de la literatura buscan la correlación entre píxeles y no la coincidencia de puntos clave. Markov es un proceso usado para determinar la probabilidad de un evento que esta dada por el evento inmediatamente anterior, se ha usado en la detección de splaicing [39], se sabe que el proceso de remuestreo altera la estadística de la imagen, en la literatura una de las técnicas mas recurrentes es la de mapas de probabilidad la cual se apoya en la regla de Bayes que busca la probabilidad de la ocurrencia de un evento dado otro. Adicional Markov y ORB no presentan problemas con la compresión JPEG ya que no usan el dominio de la frecuencia.

En el cuadro 7.1 se comparan los resultados de Amerini [43] y los propios. Amerini evaluó su técnica de SIFT en la base de datos MICC2000 y MICC600 usando en total 750, con el enfoque de detectar imágenes con copy-move. En los resultados propios se usó una mayor cantidad de imágenes (1950) enfocada en remuestreo de la base de datos MICC2000. SIFT obtuvo mejores resultados que ORB, debido a la cantidad de características que encuentra cada técnica, SIFT encuentra 256 características por punto clave y ORB encuentra 32. Además ORB encuentra muy pocos puntos claves, menos de 500.

	Exactitud	**Error**	**Sensibilidad**	**Especificidad**	**FPR**	**FNR**
Resultados de Amerini-SIFT [43]	81,3 %	18,7 %	81,6 %	92,7 %	7,3 %	18,4 %
Resultados propios de ORB	66,4 %	33,6 %	0,2 %	99,5 %	0,5 %	99,8 %

Cuadro 7.1: Comparación de la técnica ORB con SIFT en la base de datos MICC2000.

En el artículo [19] el método del espectro de energía obtuvo una exactitud de 97,2 %, mientras que la implementación propia tuvo un 90,7 % en la base de datos MICC2000, es de resaltar que las implementaciones están evaluadas en diferentes bases de datos. Qian et al. utilizaron la base de datos UCID e hicieron sus propias falsificaciones de remuestreo usando MatLab. Nguyen en [14] obtuvo una exactitud entre el 70 % y el 80 % para imágenes con rotaciones mayores a 10 grados, una exactitud del 80 % para factores de escala mayores a 1.2 y para el down-sampling obtuvo una exactitud menor al 50 %. Nguyen evaluó su técnica en 200 imágenes originales y 3400 imágenes falsificadas por él mismo, tomadas de la base de datos UCID. Las técnicas de mapas de

probabilidad y Markov implementadas en este trabajo obtuvieron una exactitud del 76,3 % y 93 % respectivamente, en la base de datos MICC2000, P-map estuvo entre los rangos descritos por el autor (70 % - 80 %) y Markov supero estos rangos. Se comparo los resultados obtenidos con Markov y los expresados en [14] ya que ambas técnicas hacen uso de la probabilidad. La comparación de los resultados de los autores de [14] y [19] con los propios pueden verse afectados ya que no se usó la misma base de datos en cada técnica, ni el mismo numero de imágenes, las demás métricas calculadas en este trabajo no se pudieron comparar con los autores ya que estos no las expresan.

Recomendaciones

La técnica de espectro de energía solo usa las filas de la imagen para hacer su desarrollo, se propone usar las columnas y las diagonales, ya que en la matriz de remuestreo 4.1 se observa que también existe periodicidad en estas direcciones, de esta forma se podrá obtener mas información sobre la periodicidad y que los picos sean mas fuertes.

El algoritmo propuesto por Amerini esta diseñado para funcionar con SIFT, al usar un nuevo extractor de puntos claves como ORB se recomienda cambiar ciertos parámetros a este algoritmo, como la distancia euclidiana por la distancia Hamming que es usada para hallar la mínima distancia entre puntos. También se recomienda utilizar RANSAC en el cluster.

Las bases de datos MICC2000 y DATASET presentan una porción muy desigual entre la cantidad de imágenes originales y falsificadas, se recomienda tener igual cantidad de imágenes alteradas y originales para evitar que la exactitud sea alta con sensibilidad o especificidad baja.

Es de resaltar que la única técnica que localiza la zona de la falsificación es ORB, para las técnicas de P-amp, Espectro de energía y Markov se propone localizar la zona mediante el analizáis individual de bloques, este analizáis debe ser realizado en bloques de diferentes tamaños ya que el área de la falsificación puede afectar los resultados.

Capítulo 8

Conclusiones

Se observó que la base de datos MICC2000 presento los mejores resultados para las técnicas de p-map, espectro de energía y Markov, siendo esta ultima la que obtuvo el mayor valor para la exactitud.

La base de datos COVERAGE obtuvo los resultados mas desfavorables, la transformación mas recurrente es la rotación y las modificaciones de escala son mínimas, lo que indica que los métodos pueden tener problemas cuando hay rotación. Por el contrario MICC2000 obtuvo los resultados mas sobresalientes, cada 8 imágenes de 14 contiene falsificaciones con solo escalado y las restantes son combinación de rotación y escalado, de tal forma que los métodos detectan mejor las alteraciones que tienen escalado.

El método de ORB presenta el error mas alto en la base da datos DATASET y MICC2000, debido a que extrae menos de 500 puntos claves y el método de Amerini esta diseñado para descartar zonas con pocas coincidencias. Adicional el algoritmo descarta zonas alteradas pequeñas.

Las imágenes con down-sampling o reducción de escala son mas difíciles de detectar por los métodos de p-map y espectro de energía, ya que estos buscan los picos producidos por la correlación que aparece al introducir nuevos píxeles, mientras que, el down-sampling quita píxeles y no se altera de forma significativa la correlación entre los píxeles.

Se notó que en algunos resultados la exactitud era cercana al 100 %, pero la sensibilidad o la especificidad eran muy bajas, esto es debido a que las bases datos presentan una desigualdad grande en la cantidad de imágenes originales y falsificadas.

Bibliografía

[1] A. C. Popescu and H. Farid, "Exposing digital forgeries by detecting traces of resampling," *IEEE Transactions on signal processing*, vol. 53, no. 2, pp. 758–767, 2005. 1, 3, 4.1, 4.1, 4.3

[2] C. P. Colombiano, "Código penal colombiano," *Código Penal Colombiano*, 2012. 2

[3] M. de Tecnologías de la Informacion y las Comunicaciones, "Seguridad y privacidad de la información, evidencia digital," 2016, [En línea]. Disponible en: https://www.mintic.gov.co/gestionti/615/articles-5482-G13-Evidencia-Digital.pdf. [Accedido: 24/07/2017]. 2

[4] M. de Tecnologías de la Información y las Comunicaciones, "Mintic busca oferta académica para formación en áreas de gestión de ti y seguridad y privacidad de la información," 2017, [En línea]. Disponible en: http://www.mintic.gov.co/portal/604/w3-article-48023.html. [Accedido: 24/07/2017]. 2

[5] T. Qiao, A. Zhu, and F. Retraint, "Exposing image resampling forgery by using linear parametric model," *Multimedia Tools and Applications*, vol. 77, no. 2, pp. 1501–1523, 2018. 3

[6] G. K. Birajdar and V. H. Mankar, "Blind authentication of resampled images and rescaling factor estimation," in *Cloud & Ubiquitous Computing & Emerging Technologies (CUBE), 2013 International Conference on*. IEEE, 2013, pp. 112–116. 3

[7] G. Birajdar and V. H. Mankar, "Blind method for rescaling detection and rescale factor estimation in digital images using periodic properties of interpolation," *AEU-International Journal of Electronics and Communications*, vol. 68, no. 7, pp. 644–652, 2014. 3

[8] G. K. Birajdar and V. H. Mankar, "Subsampling-based blind image forgery detection using support vector machine and artificial neural network classifiers," *Arabian Journal for Science and Engineering*, vol. 43, no. 2, pp. 555–568, 2018. 3

[9] B. Bayar and M. C. Stamm, "On the robustness of constrained convolutional neural networks to jpeg post-compression for image resampling detection," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 2152–2156. 3

[10] Z. Chen, Y. Zhao, and R. Ni, "Detection of operation chain: Jpeg-resampling-jpeg," *Signal Processing: Image Communication*, vol. 57, pp. 8–20, 2017. 3

[11] J. Bunk, J. H. Bappy, T. M. Mohammed, L. Nataraj, A. Flenner, B. Manjunath, S. Chandrasekaran, A. K. Roy-Chowdhury, and L. Peterson, "Detection and localization of image forgeries using resampling features and deep learning," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2017, pp. 1881–1889. 3

[12] Y. Su, X. Jin, C. Zhang, and Y. Chen, "Hierarchical image resampling detection based on blind deconvolution," *Journal of Visual Communication and Image Representation*, vol. 48, pp. 480–490, 2017. 3

[13] H. C. Nguyen and S. Katzenbeisser, "Robust resampling detection in digital images," in *IFIP International Conference on Communications and Multimedia Security*. Springer, 2012, pp. 3–15. 3

[14] H. C. Nguyen, "A machine learning based technique for detecting digital image resampling," in *Asian Conference on Intelligent Information and Database Systems*. Springer, 2016, pp. 75–84. 3, 5.1.2, 7

[15] A. Peng, H. Zeng, X. Lin, and X. Kang, "Countering anti-forensics of image resampling," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3595–3599. 3

[16] D. Vázquez-Padín and F. Pérez-González, "Exposing original and duplicated regions using sift features and resampling traces," in *International Workshop on Digital Watermarking*. Springer, 2011, pp. 306–320. 3

[17] D. Vázquez-Padín, P. Comesana, and F. Pérez-González, "An svd approach to forensic image resampling detection," in *Signal Processing Conference (EUSIPCO), 2015 23rd European*. IEEE, 2015, pp. 2067–2071. 3

[18] X. Hou, T. Zhang, G. Xiong, Y. Zhang, and X. Ping, "Image resampling detection based on texture classification," *Multimedia tools and applications*, vol. 72, no. 2, pp. 1681–1708, 2014. 3

[19] R. Qian, W. Li, N. Yu, and Z. Hao, "Image forensics with rotation-tolerant resampling detection," in *Multimedia and Expo Workshops (ICMEW), 2012 IEEE International Conference on*. IEEE, 2012, pp. 61–66. 3, 5.1.3, 7

[20] X. Feng, I. J. Cox, and G. Doërr, "Normalized energy density-based forensic detection of resampled images," *IEEE Transactions on Multimedia*, vol. 14, no. 3, pp. 536–545, 2012. 3

[21] S. Pfennig and M. Kirchner, "Spectral methods to determine the exact scaling factor of resampled digital images," in *Communications Control and Signal Processing (ISCCSP), 2012 5th International Symposium on*. IEEE, 2012, pp. 1–6. 3

[22] Y.-T. Kao, H.-J. Lin, C.-W. Wang, Y. C. Pai *et al.*, "Effective detection for linear up-sampling by a factor of fraction." *IEEE Trans. Image Processing*, vol. 21, no. 8, pp. 3443–3453, 2012. 3

[23] J. Zuo, S. Pan, B. Liu, and X. Liao, "Tampering detection for composite images based on re-sampling and jpeg compression," in *Pattern Recognition (ACPR), 2011 First Asian Conference on*. IEEE, 2011, pp. 169–173. 3

[24] G.-S. Song, Y.-I. Yun, and W.-H. Lee, "A new estimation approach of resampling factors using threshold-based peak detection," in *Consumer Electronics (ICCE), 2011 IEEE International Conference on*. IEEE, 2011, pp. 731–732. 3

[25] S.-p. Li, Z. Han, Y.-z. Chen, B. Fu, C. Lu, and X. Yao, "Resampling forgery detection in jpeg-compressed images," in *Image and Signal Processing (CISP), 2010 3rd International Congress on*, vol. 3. IEEE, 2010, pp. 1166–1170. 3

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

[27] B. Mahdian and S. Stanislav, "Detection of resampling supplemented with noise inconsistencies analysis for image forensics," in *Computational Sciences and Its Applications, 2008. ICCSA'08. International Conference on*. IEEE, 2008, pp. 546–556. 3

[28] M. Kirchner, "Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue," in *Proceedings of the 10th ACM workshop on Multimedia and security*. ACM, 2008, pp. 11–20. 3

[29] Y. Zhu, X. Shen, and H. Chen, "Copy-move forgery detection based on scaled orb," *Multimedia Tools and Applications*, vol. 75, no. 6, pp. 3221–3233, 2016. 3

[30] G. Muzaffer, O. Makul, B. Ustubioglu, and G. Ulutas, "Copy move forgery detection using gabor filter and orb," in *2016International Conference on Image Processing, Production and Computer Science, London*, 2016. 3

[31] J. D. Gutiérrez, "Informática forense giovanni zuccardi," 2006. 4

[32] H. Farid, "Image forgery detection," *IEEE Signal processing magazine*, vol. 26, no. 2, pp. 16–25, 2009. 4

[33] B. Mahdian and S. Saic, "A bibliography on blind methods for identifying image forgery," *Signal Processing: Image Communication*, vol. 25, no. 6, pp. 389–399, 2010. 4

[34] M. A. Qureshi and M. Deriche, "A bibliography of pixel-based blind image forgery detection techniques," *Signal Processing: Image Communication*, vol. 39, pp. 46–74, 2015. 4

[35] G. K. Birajdar and V. H. Mankar, "Digital image forgery detection using passive techniques: A survey," *Digital investigation*, vol. 10, no. 3, pp. 226–245, 2013. 4

[36] S. Prasad and K. Ramakrishnan, "On resampling detection and its application to detect image tampering," in *2006 IEEE International Conference on Multimedia and Expo*. IEEE, 2006, pp. 1325–1328. 4.1, 4.1

[37] W. Burger and M. J. Burge, *Principles of digital image processing: Core Algorithms*. Springer Science & Business Media, 2010. 4.1

[38] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Computer Vision (ICCV), 2011 IEEE international conference on*. IEEE, 2011, pp. 2564–2571. 4.2

[39] E.-S. M. El-Alfy and M. A. Qureshi, "Combining spatial and dct based markov features for enhanced blind detection of image splicing," *Pattern Analysis and Applications*, vol. 18, no. 3, pp. 713–723, 2015. 4.5, 7

[40] I. Amerini, L. Ballan, R. Caldelli, A. Del Bimbo, L. Del Tongo, and G. Serra, "Copy-move forgery detection and localization by means of robust clustering with j-linkage," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 659–669, 2013. 5.1.1

[41] E. Ardizzone, A. Bruno, and G. Mazzola, "Copy–move forgery detection by matching triangles of keypoints," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2084–2094, 2015. 5.2.1

[42] B. Wen, Y. Zhu, R. Subramanian, T.-T. Ng, X. Shen, and S. Winkler, "Coverageâa novel database for copy-move forgery detection," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 161–165. 5.2.1

[43] 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*, vol. 6, no. 3, pp. 1099–1110, 2011. 5.2.1, 7, **??**