



TRABAJO FIN DE GRADO
GRADO DE INGENIERÍA EN COMPUTADORES
CURSO 2016-2017

Análisis Forense de Imágenes Digitales mediante Interpolación Cromática

DANIEL POVEDANO ÁLVAREZ
DESIRÉE VETENCOURT ROSAS

Directores:

Luis Javier García Villalba
Ana Lucila Sandoval Orozco

Departamento de Ingeniería del Software e Inteligencia Artificial

FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

Los abajo firmantes autorizan a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Grado: “Análisis Forense de Imágenes Digitales mediante Interpolación Cromática”, realizado durante el curso académico 2016-2017 bajo la dirección de Luis Javier García Villalba y Ana Lucila Sandoval Orozco en el Departamento de Ingeniería del Software e Inteligencia Artificial, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Periodo de embargo:

■ 12 meses

Daniel Povedano Álvarez

Desirée Vetencourt Rosas

Luis Javier García Villalba

Ana Lucila Sandoval Orozco

Agradecimientos

Primero, queremos agradecer a Luis Javier García Villalba y a Ana Lucila Sandoval Orozco, los directores de este trabajo, todo el apoyo que nos han proporcionado para la realización del mismo.

Segundo, agradecer los miembros de nuestra familia, a nuestros padres y amigos que nos apoyado y animado durante todo el trabajo, que además, nos han ayudado aportando las fotos necesarias para el proyecto, así como la oportunidad de poder realizar el Grado de Ingeniería en Computadores.

Por último, agradecer a todos aquellos equipos de desarrolladores que han desarrollado las herramientas y documentación necesarias para la comunidad y que han servido como soporte para la realización de este trabajo

Resumen

Actualmente los dispositivos móviles equipados con cámara juegan un papel importante en la sociedad debido a su frecuente uso. Esto hace que la identificación del dispositivo móvil adquiera importancia en el análisis forense de imágenes digitales. El objetivo de este trabajo es diseñar un algoritmo para identificar la marca y el modelo de la cámara que ha generado una imagen dada. El trabajo gira en torno a los algoritmos de interpolación cromática aplicados en el proceso de formación de la imagen. En el algoritmo se estiman los coeficientes de interpolación de la matriz de filtros de color para extraer las características que servirán como huella de la cámara que capturó la escena. Los resultados de los experimentos realizados demuestran que los algoritmos de interpolación cromáticas son efectivos para identificar modelos entre distintos fabricantes con diferentes conjuntos de características.

Palabras clave

Análisis Forense de Imágenes, Dispositivos móviles, Extracción de Características, Identificación de la Fuente, Interpolación Cromática, Matriz de Filtros de Color, SVM.

Abstract

Currently, mobile devices with digital cameras play an important role in society because they are widespread and frequently used. This has led to a number of new legal and criminal issues related to the field of the multimedia, such as digital imaging, which makes the identification of the cellphone an important step in the digital forensic branch of the image. The objective of this research is to design an algorithm to solve the problem of identification of the camera model, based on the algorithms applied in the process of image formation through the estimation of interpolation coefficients of the color filter matrix (CFA), which suggests a combination of algorithms based on the identification of the source of the cell phone image strategy. This way, it identifies the atypical mobile models from the training sets. The experimental results show that our methods present a precise identification of the source for different sets of characteristics within the cellphones in the training sets.

Keywords

Color Filter Array, Demosaicing, Features Extraction, Image Forensics Analysis, Mobile Device, Source Camera Identification, SVM.

Lista de Acrónimos

CCD	Charge Coupled Device.
CFA	Color Filter Array.
CMOS	Complementary Metal Oxide Semiconductor.
CYMK	Cyan, Magenta, Yellow y Key.
DFT	Discrete Fourier Transform
DPI o DSP	Digital Image Processor.
EXIF	Exchangeable Image File Format.
EM	The Expectation Maximization.
GA	Genetic Algorithm.
IQM	Image Quality Metrics.
ISO	International Organization for Standardization.
JPEG	Joint Photographic Experts Group.
LRD	Radial Distortion of the Lens.
PDF	Portable Document Format.
PNG	Portable Network Graphics.
RGB	Red-Green-Blue.
SNR	Signal to Noise Ratio.
SFFS	Sequential Forward Featured Selection.
SFS	Sequential Floating Search.
SPN	Sensor Pater Noise.
SVM	Support Vector Machine.

ÍNDICE

1. INTRODUCCIÓN	1
1.1. MOTIVACIÓN	1
1.2. CONTEXTO	2
1.3. OBJETO DEL PROYECTO	3
1.4. ESTRUCTURA DE LA MEMORIA	4
1.5. PLANIFICACIÓN DEL TRABAJO	5
1.6. DIVISIÓN DEL TRABAJO	6
2. ADQUISICIÓN Y PROCESAMIENTO DE IMÁGENES DIGITALES	7
2.1. FILTROS DE COLOR	8
2.1.1. FILTRO DE BAYER	9
2.2. PROCESADOR DIGITAL DE IMAGEN	10
2.3. MODELO DE INTERPOLACIÓN CROMÁTICA	12
2.4. ALGORITMOS DE INTERPOLACIÓN CROMÁTICA	13
2.4.1. ALGORITMOS DE INTERPOLACIÓN NO ADAPTATIVOS	14
2.4.2. ALGORITMOS DE INTERPOLACIÓN ADAPTATIVOS	15
2.4.3. REDUCCIÓN DEL RUIDO DEL SENSOR	15
2.4.4. FILTRO LINEALES DE SUAVIZADO	17
2.4.5. DETECCIÓN DE BORDES.....	18
3. TÉCNICAS DE ANÁLISIS FORENSE EN IMÁGENES.....	19
3.1. TÉCNICAS BASADAS EN LOS METADATOS	19
3.2. TÉCNICAS BASADAS EN EL HARDWARE DEL DISPOSITIVO.....	20
3.3. TÉCNICAS BASADA EN LA ABERRACIÓN CROMÁTICA.....	21
3.4. TÉCNICAS BASADAS EN LAS IMPERFECCIONES DEL SENSOR	22
3.4.1. DEFECTOS DEL PIXEL	22
3.4.2. PATRÓN DEL RUIDO DEL SENSOR	23
3.5. TÉCNICAS BASADAS EN EL SOFTWARE DEL DISPOSITIVO	25
3.6. CARACTERÍSTICAS DE LA IMAGEN	25
3.7. MÉTRICAS PARA CADA BANDA DE COLOR DE VARIOS MOMENTOS GENERALIZADOS DE DEMOAISING RESIDUAL.....	27
4. CONTRIBUCIÓN	29
4.1. REGULARIDADES DE PROCESAMIENTO BASADOS EN LOS COMPONENTES DEL DPI.....	29
4.2. CONCEPTOS GENERALES.....	34
4.2.1. ESPERANZA MAXIMIZACIÓN	34
4.2.2. DETECCIÓN DE INTERPOLACIÓN	35
4.2.3. ESTRATEGIAS DE SELECCIÓN DE CARACTERÍSTICAS.	36
4.3. ESPECIFICACIÓN DEL ALGORITMO	38
4.3.1. EXTRACCIÓN DE LOS PESOS DE LOS COEFICIENTES DE INTERPOLACIÓN.....	39
4.3.2. DETECCIÓN DEL TIPO DE INTERPOLACIÓN.....	42
4.3.3. SELECCIÓN DE CARACTERÍSTICAS	43
4.4. HERRAMIENTAS DE DESARROLLO.....	44
5. EXPERIMENTOS Y RESULTADOS	45
5.1. ANÁLISIS DE MÓVILES DEL MISMO FABRICANTE Y DIFERENTES MODELOS	46
5.2. EXPERIMENTO DE DETECCIÓN DE TIPOS DE INTERPOLACIÓN.....	49
5.3. EVALUACIÓN DE LOS PESOS DE LOS COEFICIENTES DE INTERPOLACIÓN	54
6. CONCLUSIONES Y TRABAJO FUTURO.....	59
6.1. CONCLUSIONES.....	59
6.2. TRABAJO FUTURO.....	60
7. INTRODUCTION	63

7.1. MOTIVATION.....	63
7.2. OBJECTIVES	63
7.3. WORK SCHEDULE	64
8. CONCLUSIONS AND FUTURE RESEARCH	65
8.1. CONCLUSIONS.....	65
8.2. FUTURE RESEARCH.....	66
REFERENCIAS	67

ÍNDICE DE TABLAS

Tabla 1.1. Fases del proyecto	5
Tabla 3.1: Técnicas de identificación de la fuente respecto al hardware.....	24
Tabla 3.2: Comparación de las técnicas de identificación de la fuente basadas en la configuración CFA y las funciones de interpolación de color.	33
Tabla 5.1: Distintos Experimentos Implementados.	45
Tabla 5.2 Resultados Experimento P1_A	46
Tabla 5.3 Resultados Experimento con 8 dispositivos	47
Tabla 5.4 Resultados Experimento con 11 dispositivos	48
Tabla 5.5 Resultados Experimento 4 dispositivos	48
Tabla 5.6 Resultados Experimento Zonas Suaves.....	49
Tabla 5.7 Resultados de la evaluación de los pesos de los coeficientes de interpolación sin SFFS.....	54
Tabla 5.8 Resultados Experimento Esperanza Maximización 2 con SFFS.	54
Tabla 5.9: Resultados Experimento Esperanza Maximización 3.	55
Tabla 5.10: Resultados Experimento Esperanza Maximización Experimento Apple.....	56
Tabla 5.11: Resultados Experimento Esperanza Maximización EXPERIMENTO4_50. .	56
Tabla 5.12: Resultados Experimento Esperanza Maximización EXPERIMENTO5.	57

ÍNDICE DE FIGURAS

Figura 2.1: Proceso de adquisición de imagen en dispositivos móviles.....	7
Figura 2.2: Filtro de Bayer RGB.....	9
Figura 2.3: Funciones del procesador digital de imagen.....	13
Figura 3.1: Patrón del sensor jerarquía del ruido.....	16
Figura 3.2: Ejemplo de aberración cromática.	22
Figura 4.1: Técnica del algoritmo EM.....	35
Figura 4.2 Tipo de frecuencia de la segunda derivada de una imagen interpolada.....	36
Figura 4.3: Esquema del algoritmo SFFS.....	38
Figura 5.1: Ejemplo Samsung J500 sin Zoom	50
Figura 5.2: Ejemplo Samsung J500 con Zoom	50
Figura 5.3: Ejemplo Lenovo sin Zoom.....	51
Figura 5.4: Ejemplo Lenovo con Zoom.....	51
Figura 5.5: Ejemplo Iphone 6 sin Zoom	52
Figura 5.6: Ejemplo Iphone 6 con Zoom	52
Figura 5.7: Ejemplo Samsung Tab 2 sin Zoom	53
Figura 5.8: Ejemplo Samsung Tab 2 con Zoom.....	53

1. INTRODUCCIÓN

1.1. Motivación

Actualmente los dispositivos móviles evolucionan con gran rapidez para adaptarse a las necesidades de los usuarios y también del mercado produciendo un incremento de ventas de móviles de diferentes compañías y modelos. Uno de los componentes que buscan los usuarios en los dispositivos móviles es una cámara fotográfica de alta calidad integrada. Esto obliga a las compañías fabricantes de este tipo de dispositivos a mejorar la calidad de la cámara incorporada en las mismas sin incrementar en gran medida los costes del mismo. Como consecuencia, el usuario lleva consigo el dispositivo móvil siempre a mano, lo que le permite captar cualquier tipo de escena en cualquier momento [JWWABA16]. Asimismo, los usuarios pueden compartir automáticamente las fotos y vídeos generados por estos dispositivos en las redes sociales a las que está suscrito.

Dada la cantidad de tiempo que un usuario gasta usando su dispositivo móvil, éste se ha convertido en el primer dispositivo de generación de fotografías. Estas imágenes son usadas actualmente en los noticieros, programas de entretenimiento, etc. Adicionalmente, estas fotografías pueden ser utilizadas como pruebas en investigaciones de procesos judiciales, que en algunos casos son piezas fundamentales de la evidencia de un delito [AZ06] [WY06]

Esto sumado al valor de la información y al uso que se le da a ésta por parte de compañías privadas, empresas de publicidad, organismos de gobierno e individuos en general, hacen que el análisis forense multimedia adquiera una gran importancia en la actualidad [SB13].

1.2. Contexto

El presente Trabajo Fin de Grado se enmarca dentro de un proyecto de investigación titulado RAMSES aprobado por la Comisión Europea dentro del Programa Marco de Investigación e Innovación Horizonte 2020 (Convocatoria H2020-FCT-2015, Acción de Innovación, Número de Propuesta: 700326) y en el que participa el Grupo GASS del Departamento de Ingeniería del Software e Inteligencia Artificial de la Facultad de Informática de la Universidad Complutense de Madrid (Grupo de Análisis, Seguridad y Sistemas, <http://gass.ucm.es>, grupo 910623 del catálogo de grupos de investigación reconocidos por la UCM).

Además de la Universidad Complutense de Madrid participan las siguientes entidades:

- Treelogic Telemática y Lógica Racional para la Empresa Europea SL (España)
- Ministério da Justiça (Portugal)
- University of Kent (Reino Unido)
- Centro Ricerche e Studi su Sicurezza e Criminalità (Italia)
- Fachhochschule für Öffentliche Verwaltung und Rechtspflege in Bayern (Alemania)
- Trilateral Research & Consulting LLP (Reino Unido)
- Politecnico di Milano (Italia)
- Service Public Fédéral Intérieur (Bélgica)
- Universität des Saarlandes (Alemania)
- Dirección General de Policía - Ministerio del Interior (España)

1.3. Objeto del Proyecto

El presente Trabajo Fin de Grado (TFG) tiene los siguientes objetivos:

- Comprender la importancia del análisis forense multimedia en la actualidad. Más concretamente en los procesos involucrados en la generación de imágenes de dispositivos móviles.
- Realizar un estudio de los trabajos relacionados con las técnicas de identificación de la fuente de imágenes haciendo énfasis en las técnicas basadas en los algoritmos de interpolación utilizados en el procesador digital de la imagen DPI (*Digital Image Processor*).
- Analizar las diferentes técnicas de extracción de características aplicadas a imágenes digitales con compresión JPG.
- Estudiar los diferentes sistemas de clasificación existentes detallando los sistemas supervisados más utilizados en investigaciones relacionadas con dispositivos móviles.
- Diseñar un algoritmo de identificación de la fuente que combine la extracción de características con la información que aporta el procesador digital de imagen durante la generación de la imagen.
- Diseñar un algoritmo de selección de las características más relevantes del procesador digital de imagen, con el objeto de mejorar los resultados de la clasificación.
- Realizar experimentos con imágenes de dispositivos móviles para evaluar la efectividad del algoritmo diseñado.

1.4. Estructura de la Memoria

El resto del trabajo está organizado en 6 capítulos con la estructura que se comenta a continuación.

En el capítulo 2 se presenta una reseña histórica de la fotografía, el proceso de generación de una imagen de dispositivo móvil, los tipos de sensores utilizados en una cámara de dispositivo móvil y la descripción de los principales sistemas de metadatos en imágenes digitales.

En el capítulo 3 presenta tipos de técnicas de uso del DPI que se pueden realizar sobre una imagen digital según el objetivo de la identificación. En el capítulo se presentan los trabajos relacionados con las técnicas de identificación en imágenes digitales.

En el capítulo 4 se presenta el algoritmo de identificación propuesto en este trabajo basado en el DPI y la elección de la mejor característica SFFS (*Sequential Forward Floating Selection*). Seguidamente se realiza una evaluación del mismo para analizar su grado de efectividad.

En el capítulo 5 se presentan las principales conclusiones extraídas de este trabajo y las líneas de trabajo futuro.

En los capítulos 6 se realiza un resumen en inglés de la introducción y las conclusiones del trabajo.

1.5. Planificación del Trabajo

El proyecto se ha desarrollado en 3 fases: Definición, Ejecución y Documentación del Proyecto. Las actividades realizadas en cada una de estas fases se presentan en la Tabla 1.1.

Nombre de tarea	Inicio	Fin
• Definición del proyecto	14/11/16	27/01/17
- Reuniones semanales de seguimiento con los tutores.		
- Estudio de los tipos de identificación de imágenes.		
- Estudio de las técnicas de identificación de imágenes aplicadas a las imágenes obtenidas en dispositivos móviles.		
- Definición del proyecto		
• Ejecución del Proyecto	20/02/17	26/05/17
- Especificación de requisitos		
- Diseño		
- Implementación		
- Pruebas		
- Control		
• Documentación	20/12/17	05/06/17
- Generación de documentación del proyecto		
- Preparación de la memoria		

Tabla 1.1. Fases del proyecto

Durante la primera fase se establecieron los objetivos y alcance del Trabajo de Fin de Grado, las reuniones con el equipo de tutores y el seguimiento del trabajo realizado durante la elaboración del trabajo. Posteriormente, durante la fase de Ejecución, se desarrolló el proyecto definido en la etapa anterior. Esta fase está compuesta por las siguientes etapas: Especificación de requisitos, diseño, implementación y pruebas. Durante la realización de esta fase, se realizaron actividades de seguimiento y control del avance del proyecto para

llevar un control y seguimiento de las actividades realizadas.

Finalmente, en la fase de documentación, se realizó toda la documentación necesaria para la elaboración del Trabajo Fin de Grado. Esta fase se ha realizado en conjunto con las dos fases anteriores.

1.6. División del Trabajo

En la primera etapa del proyecto, realizamos un curso de PYTHON, a través de la plataforma educativa de Coursera, en conjunto con manuales de la Universidad Complutense de Madrid sobre este tema, para obtener todos los conocimientos necesarios a cerca de este lenguaje de programación.

Para la realización de las primeras fases de la memoria relacionado al estado del arte y la motivación, realizamos búsqueda de información de trabajos relacionados con el análisis forense, procesos de creación de imágenes y de la incidencia en la sociedad actual. Así mismo, para la fase de implementación con investigación de diferentes artículos relacionados a las técnicas de hardware y la parte que aporta este proyecto que son las técnicas de software.

Después de la realización de la investigación de las diferentes técnicas y estudio de las diferentes implementaciones de algoritmos estadísticos se procedió al desarrollo de los diferentes algoritmos de identificación de la imagen: Extracción de los pesos de los coeficientes de interpolación cromática, detección de interpolación de picos y selección de características más relevantes, con sus debidas pruebas

2. ADQUISICIÓN Y PROCESAMIENTO DE IMÁGENES DIGITALES

En el proceso de creación de una imagen las características de este, dependen de los componentes utilizados por el fabricante en la elaboración de un teléfono móvil. Por ejemplo, el tipo de sensor y/o filtro utilizado al tomar una fotografía produce un patrón único de características o información sobre un teléfono móvil [HTRC16]. Cada fabricante utiliza diferentes componentes en la fabricación de una cámara fotográfica digital. Sin embargo, la estructura de una cámara siempre es la misma, como se observa en la Figura 2.1. Esta información de patrones puede ser utilizada como una firma, debido a las diferencias presentes en los distintos componentes utilizados por los fabricantes en las cámaras digitales y al proceso de generación de las imágenes en las mismas [JWWABA16].

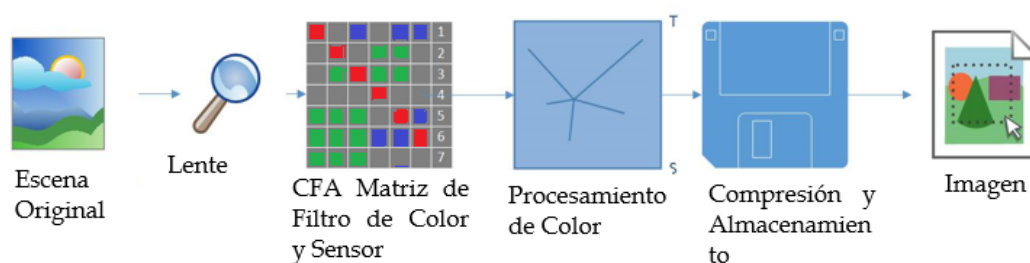


Figura 2.1: Proceso de adquisición de imagen en dispositivos móviles.

Una cámara digital está compuesta por: un sistema de lentes, diferentes filtros para la obtención del color como la matriz de filtro de color (CFA del inglés, *Color Filter Array*), un sensor y un procesador digital de imagen (DPI). A continuación, se procesa la imagen eliminando el ruido y las aberraciones con una serie de algoritmos. Finalmente, la imagen generada por el procesador se comprime. Al momento de la creación de la imagen se inserta en la foto una cabecera de información de archivo extendida, denominada metadatos EXIF (del inglés, *Exchangeable Image File Format*), en ella se almacena toda la

información relativa a la foto capturada por la cámara del dispositivo móvil. Esta información incluye datos sobre la cámara, condiciones de captura de la escena, datos de ubicación, fecha y hora de creación, etc. Los metadatos EXIF se utiliza en los modelos de compresión el JPEG (del inglés, *Joint Photographic Experts Group*) y PNG (del inglés, *Portable Network Graphics*), que son los soportes de modelo de color RGB y CYMK para la representación de la información de la imagen. Sin embargo, la información que aportan los metadatos EXIF no es fiable ya que se pueden modificar fácilmente [JWWABA16].

Los modelos de composición de las fotos utilizados en los dispositivos móviles dependen del filtro que tenga el sensor digital de la cámara sea CCD (del inglés, *Charge Coupled Device*) o CMOS (del inglés, *Complementary Metal Oxide Semiconductor*) con estos la información de luminosidad se obtiene para cada pixel de color.

2.1. Filtros de Color

Se presentan dos modelos de colores en la composición de las imágenes plasmadas en las pantallas de los dispositivos, proyectores, escáneres y cámaras uno de los modelos es el RGB (del inglés, *Red, Green and Blue*) y el modelo de color CYMK (del inglés, *Cyan, Magenta, Yellow y Key*) [ME11].

En los dispositivos que utilizan la composición RGB, los colores se generan con respecto al incremento de las luces emitidas por los diferentes algoritmos de interpolación que generan el color. Por ejemplo, el color negro se presenta (0,0,0) por su ausencia de color, el blanco (255,255,255) y el gris se forma cuando los componentes son iguales para cada uno de los elementos de 8 bits del RGB (rojo, verde, azul). De esta forma se obtienen imágenes de 32 bits para los tres canales, 8 bits de información de cada pixel.

Los modelos de colores CMYK se basan en la sustracción de los elementos

que lo generan combinándose entre sí todos los colores y absorbiendo la luz para generar el negro. De esta forma se le asignan valores de porcentajes para cada pixel de color [ME11].

2.1.1. Filtro de Bayer

Los dispositivos electrónicos aplican diferentes transformaciones para la creación de una imagen una de estas técnicas es la recolección de luz, que emplean una transformación analógica-digital. Además de realizar esta transformación se aplica otra para el coloreado de la imagen llamada transformación de Bayer. Utilizando la transformación de Bayer los colores se obtienen a partir del filtro de Bayer del sensor de imagen con el modelo del filtro RGB. Sin embargo, cada fabricante utiliza un filtro Bayer distinto en sus dispositivos.

El filtro de Bayer consiste de un componente de color rojo y un componente de color azul y dos componentes de color verde en una celda 2x2, ya que el ojo humano percibe más el color verde con respecto a otros colores por lo tanto las celdas del filtro Bayer deja pasar más las longitudes de ondas de color verde correspondiente al 50 %, que del azul que supone 25% y el rojo otro 25%. La Figura 2.3 muestra una representación del filtro de Bayer.

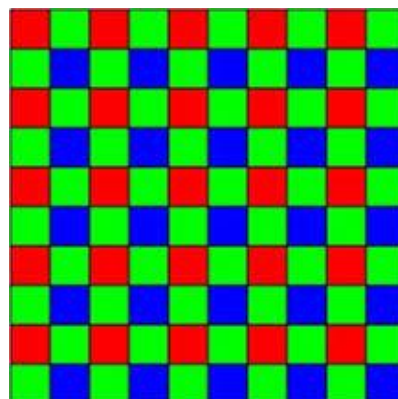


Figura 2.2: Filtro de Bayer RGB

2.2. Procesador Digital de Imagen

El DPI realiza diferentes transformaciones digitales básicas, como interpolación cromática, reducción de ruido, detección de bordes, corrección gamma, entre otros. Estas transformaciones se utilizan para la formación de la imagen a partir de los datos que se obtienen del sensor. Cada fabricante aplica un algoritmo diferente en las operaciones en el proceso de creación de una imagen.

La interpolación cromática se encarga de la reconstrucción de la imagen en colores mediante las muestras obtenidas de los sensores. El resultado obtenido puede variar entre los diferentes fabricantes, ya que, en el procesado de la imagen suelen usar diferentes algoritmos para la interpolación.

De todas las etapas que realiza el DPI, la principal es la interpolación cromática, que se encarga de reconstruir la imagen a partir de los datos obtenidos del sensor. Otra de las funciones del DPI es la reducción del ruido del sensor. Este ruido tiene las siguientes características:

- Depende de la temperatura, a temperaturas altas es más elevado el ruido del sensor.
- Es de naturaleza aleatoria.
- Sigue una distribución denominada normal o Gaussiana.
- Está relacionada con el nivel que se aplique de la ISO (del inglés, *International Organization for Standardization*) basado en la sensibilidad de la fotografía. Al aumentar la ISO aumenta la sensibilidad del sensor con la consiguiente reducción del nivel de señal a ruido o SNR (*Signal to Noise Ratio*).

Para la reducción del ruido el DPI utiliza varios tipos de filtros que permiten manejar de manera distinta el ruido de luminancia y de crominancia. Debido a

esta reducción y a la interpolación cromática la imagen puede ver afectada su nitidez. Esto obliga al DPI a usar algoritmos de corrección de nitidez o *sharpening* para corregir la nitidez. Para llevar a cabo esta corrección, los algoritmos tienen que ser capaces de detectar los bordes y contornos de la imagen ya que son las regiones más afectadas de la imagen por esta pérdida de nitidez.

Algunos ejemplos de regularidades del procesamiento son [JWWABA16]:

- **Balanceo de Blancos:** Un componente importante en el procesamiento de imágenes es el equilibrio de blancos. El algoritmo de equilibrio de blancos garantiza la reproducción del color mediante la corrección de las imágenes de la fuente de luz [AGRCSO+14].
- **Corrección Gamma:** La no linealidad de luminancia introducida por muchas imágenes en los dispositivos, se pueden describir mediante una simple operación puntual conocida como corrección Gamma. Las huellas de estas operaciones han sido utilizadas para la detección forense de la manipulación de imágenes [AGRCSO+14].
- **Configuración del CFA:** Es un conjunto de filtros de colores del sensor, cada pixel de color primario es interpolados para la estimación de los colores faltantes, este proceso es denominado interpolación cromática en inglés *demosaising*. La configuración CFA más comúnmente utilizada es el filtro Bayer en la que se alterna rojo (R) y verde (G) para filas impares y filtros alternos verde (G) y azul (B) para filas pares.
- **Interpolación cromática:** Es el proceso de reconstrucción de la imagen mediante las muestras cromáticas adquiridas del sensor.
- **Compresión de la imagen:** Es el proceso de reducción del tamaño de la codificación, la compresión puede ser con pérdida y sin pérdida, la

compresión sin pérdida se encuentran los sistemas adaptativos, no adaptativos y semi adaptativos.

2.3. Modelo de Interpolación Cromática

El método de interpolación cromática o *demosaising CFA* reconstruye la imagen usando los datos recogidos por el sensor de imagen recubierto por el filtro de color (RAW). Esta reconstrucción tiene mucho impacto en el resultado final [WAW+16]. Incluso se podría decir que una de las partes más importantes en la fotografía de telefonía móvil es el tipo de algoritmo utilizado para la reconstrucción final, siendo los responsables de que la cámara genere una imagen de calidad. Sin embargo, el hardware también es importante, pero no serviría de mucho sin un buen algoritmo de interpolación cromática [WAW+16].

A partir de las muestras recogidas por el filtro, el sensor detecta el componente RGB muestreado en las diferentes localizaciones de los píxeles según el patrón CFA. La salida que se obtiene del sensor es una imagen en mosaico ya que solo hay un elemento de cada color en cada píxel. Para la reconstrucción de la imagen, los colores que faltan en cada uno de los píxeles son interpolados usando los datos muestreados en el área cercana a cada píxel usando algunos de los diferentes algoritmos de interpolación. Esta etapa en el procesamiento de la imagen es muy importante ya que es la que se encarga de la formación de 2/3 de la imagen final. El patrón CFA más utilizado es el patrón Bayer.

Después de este proceso de interpolación, el DPI realiza un post-procesado como la corrección gamma, reducción de ruido, detección de bordes, balanceo de blancos etc. para finalmente almacenar la imagen en un formato predefinido [WTZ+15].

Los pasos realizados por el DPI para mejorar la calidad de la foto, se pueden

observar en la Figura 2.3.

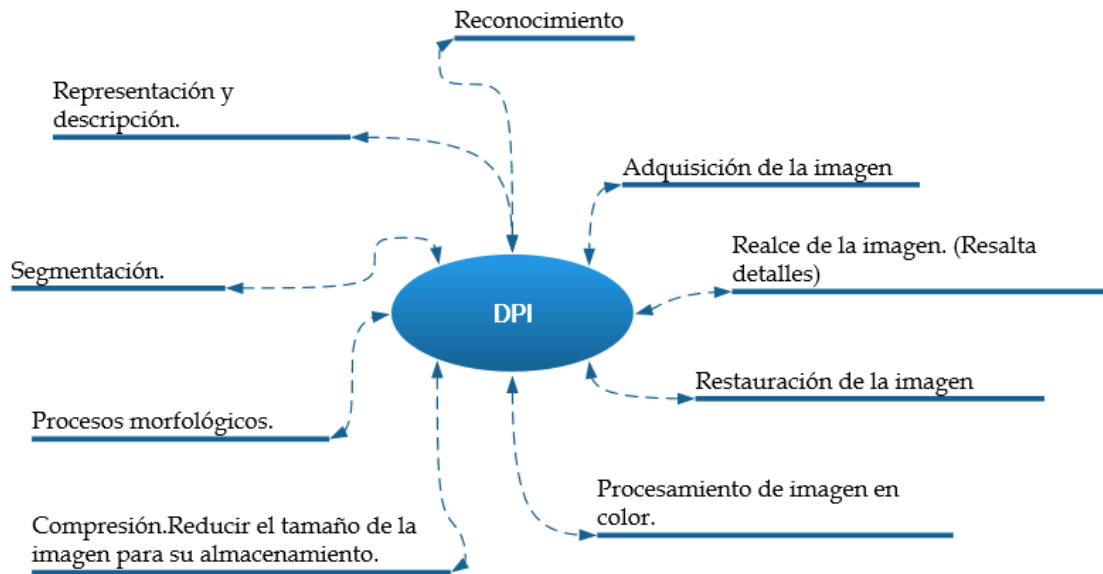


Figura 2.3: Funciones del procesador digital de imagen

Además de las características observadas en la Figura 2.3, el algoritmo ideal del DPI debe evitar la aparición de impurezas de color en la imagen como: aliasing, bordes de escalera (cambios abruptos de intensidad en un número de píxeles vecinos), entre otros, en otras palabras, conservar al máximo posible la resolución de la imagen.

2.4. Algoritmos de Interpolación Cromática

Hoy en día las cámaras de los dispositivos móviles usan un filtro de color simple como el CFA para la recuperación de los valores perdidos. Los algoritmos de interpolación utilizados para mejorar las imágenes generadas por las cámaras de los dispositivos móviles se clasifican en algoritmos adaptativos y no adaptativos.

2.4.1. Algoritmos de Interpolación no Adaptativos

Los algoritmos no adaptativos utilizados para la reconstrucción de las imágenes son los siguientes [int]:

- **Interpolación no adaptable de un solo canal:** Del inglés (*Non- Adaptive single-channel interpolation*). Es una de las interpolaciones más usadas, consiste en interpolar cada canal de color por separado usando técnicas estándar como interpolación del vecino más cercano, interpolación bilineal [Eri07].
- **Interpolación por aproximación:** Es una de las técnicas más antiguas (y más básica) ya que solo tiene en cuenta un solo píxel (el más cercano al punto interpolado) [Uni14].
- **Interpolación bilineal:** Es una mejora de la técnica anterior, promediando en este caso 4 píxeles adyacentes (2x2). Se hace el promedio ponderado de estos 4 píxeles y se obtiene el valor interpolado. Esta técnica emplea mayor tiempo de cómputo que la anterior, pero se obtiene una imagen más suavizada [Eri07] [Uni14]
- **Interpolación bicúbica:** Es similar al anterior, pero con la diferencia de que se toma ahora los 16 píxeles adyacentes (4x4). Este método está muy bien valorado ya que se presenta como una buena combinación entre tiempo y resultados [Eri07] [Uni14].
- **Interpolación en escalera:** Se basa en la interpolación bicúbica pero con la diferencia de que se va interpolando en incrementos de un 10% en cada etapa con respecto al anterior [Cam13].
- **Interpolación S-Spline:** Determina el color de un píxel «desconocido» basándose en la totalidad de colores de la imagen, a diferencia que los métodos anteriores.

2.4.2. Algoritmos de Interpolación Adaptativos

Los algoritmos adaptativos utilizados para la reconstrucción de las imágenes son los siguientes:

- **Interpolación basada en los bordes:** Basándose en el supuesto que los canales de color tienen una textura similar, pueden usarse varios detectores de bordes.
- **Interpolación basada en los tonos constantes:** el color no cambia abruptamente dentro de un pequeño vecindario de píxeles.
- **Interpolación basada en los bordes de la matriz:** Es una combinación entre la técnica de interpolación de bordes y la interpolación basada en tonos constantes, el color se interpola como en el enfoque de interpolación de tonos constantes, sin embargo, en esta técnica, se interpola la matriz en base a las direcciones de los bordes como en interpolación orientada a los bordes.
- **Uso de laplaciano para mejorar:** Usa los gradientes de segundo orden de los canales rojo-azul para realzar el canal verde. (LaRoche-Prescott) [Eri07, Cam13].

2.4.3. Reducción del Ruido del Sensor

Las imágenes procesadas por el DPI presentan un ruido que aparecerá en la imagen final, varios estudios indican que el patrón de ruido es distinto en los diferentes modelos de cámaras digitales [JWWABA16]. El ruido del sensor se puede clasificar como se muestra en la siguiente Figura 3.1:



Figura 3.1: Patrón del sensor jerarquía del ruido.

La distorsión radial de la lente es el tipo más grave con la señal ruido se han realizados estudios que prueba la huella dactilar con otros que también tienen huella dactilar [JWWABA16].

Existen dos tipos de ruidos, el ruido gausseano y ruido sal y pimienta, la primera se basa en el ruido que proviene de los componentes electrónicos, que afecta a toda la imagen, debido a que la intensidad de los pixeles se ve alterada, su función probabilística esta expresada por la curva gaussiana definida como μ es la media y σ es la desviación típica de la variable aleatoria.

El ruido sal y pimienta el valor que toma los pixeles son aleatorios de color blanco o de color negro, este ruido es producido por el mal funcionamiento de pixeles en los sensores de una cámara, en situaciones de memoria defectuosas en el hardware o por transformaciones de un canal ruido.

Existen distintos tipos de algoritmos que permiten la reducción del ruido y/o efectos espurios que pueden presentarse en una imagen a consecuencia del proceso de captura, digitalización y transmisión. Su utilización es normalmente necesaria antes de la aplicación de un detector de bordes.

2.4.4. Filtro Lineales de Suavizado

Este método consiste en aplicar a la imagen original una máscara que representa el filtro de paso bajo, es decir, suavizando la imagen [GMR10]. La convolución en la máscara gaussiana tiene el valor de cada pixel relacionada con los valores de sus respectivos vecinos, suavizando cada pixel al valor medio.

Los filtros pueden dividirse en lineales que a su vez se clasifican en filtro de paso bajo, filtros de paso alto, filtros de pasa banda y los filtros no lineales se clasifican en filtros máximos, mínimos y filtros de mediana.

- **Filtros Lineales:** Se realiza una convolución entre la imagen que se desea filtrar y la máscara, sin embargo, presenta inconveniente en la nitidez de la imagen ya que se produce un enturbiamiento.
- **Filtros no Lineales:** Los filtros no lineales se dividen en filtro mínimos, máximos y de mediana, estos filtros son encargados para resaltar los componentes para la eliminación de ruido.
- **Filtro de media y Filtro Gaussiano:** En estos los filtros se calcula la mediana de la vecindad del pixel correspondiente, de esta manera, se homogenizan los pixeles diferentes con sus vecinos. Otro método utilizado es el anisotrópico, implementando una ecuación diferencial parcial de suavizado, similar a la ecuación de calor, pero con un coeficiente de difusión es diseñado para la detección de bordes que gracias a esto el ruido puede ser eliminado sin desdibujar los bordes de la imagen [ALMB09].
- **Filtro de la Mediana:** Es otro enfoque para la eliminación del ruido, se basa en el cálculo de la mediana de los valores ordenados de los pixeles vecinos seleccionando el del medio, de esta manera, poder visitar cada pixel y reemplazarlo por la mediana de sus pixeles vecinos, es muy eficiente para

los casos de ruidos sal y pimienta.

2.4.5. Detección de Bordes

Otro procedimiento para la obtención de las huellas que deja la aberración de la lente es la detección de bordes, basada en el operador Canny. Uno de los métodos relacionados con la detección de bordes es el uso de la primera derivada parciales, tomando el valor de cero en todas las regiones donde no varía la intensidad, de esta manera poder encontrar la magnitud e intensidad del mejor borde. Este algoritmo además de obtener el gradiente para el cálculo de la magnitud y orientación del vector en cada pixel, también aplica una función de hipótesis basada en dos umbrales con el fin de eliminar contornos falsos en la imagen, suavizándolos mediante el promedio de los valores de intensidad de los pixeles vecinos [Reb07].

3. TÉCNICAS DE ANÁLISIS FORENSE EN IMÁGENES

Existen diferentes técnicas para la identificación de una imagen, en este punto se van a enumerar y describir cada una de ellas. Según [JWWABA16] se pueden definir tres grupos de técnicas: técnicas basada en la información aportada de los metadatos, técnicas basadas en el hardware del dispositivo y las técnicas basadas en el software del dispositivo.

Las técnicas de análisis basadas en metadatos se basan en información relacionada con la captura de la imagen y los datos de encabezados de la imagen.

Por otro lado, las técnicas basadas en el hardware del dispositivo estudian el rastro que dejan los componentes físicos del dispositivo móvil. Esta categoría a su vez se subdivide en: aberraciones de las lentes e imperfecciones del sensor.

Por ultimo las técnicas basadas en el software del dispositivo están relacionadas con el rastro que dejan los algoritmos que realiza el DPI en la generación de una imagen. Esta categoría se divide en dos grupos: Técnicas que realizan estadísticas de procesamiento y técnicas que analizan las regularidades del procesamiento. Esta última es el objeto de investigación del presente trabajo ya que son tareas realizadas por el DPI.

3.1. Técnicas basadas en los Metadatos

Esta técnica es una de las más sencillas, en el cual, se extraen los datos de la marca, el modelo, sistema operativo del dispositivo móvil, entre otros, estos datos dependerán del fabricante ya que determina los datos insertar como metadatos en la captura de la imagen.

Este método es bastante vulnerable a factibles cambios por terceros, es decir,

si hay elementos añadidos o retirados que no correspondan al momento de la toma. El hallazgo de estas manipulaciones es uno de los aspectos más interesantes a la hora de hacer un análisis forense de una imagen.

3.2. Técnicas basadas en el Hardware del Dispositivo

Los artefactos relacionados con el hardware son las imperfecciones presentes en la lente y en el sensor. Estas imperfecciones pueden dejar huellas del dispositivo en la imagen generada.

Las imperfecciones de la lente pueden ser generadas por la iluminación, distorsión radial de la lente, aberración y "*vignetting*". Las probabilidades de que estas imperfecciones sean usadas para la identificación de un dispositivo móvil son mínimas ya que los fabricantes realizan procesos para reducir dichas imperfecciones de la lente. Los problemas restantes aparecen por las propias características del componente [JWWABA16] Algunas de estas características son:

- **Características de los parámetros:** Son aquellas características que, debido a los parámetros estimados, solo son capaces de captar las huellas geométricas o cromáticas: La similitud de dichos valores es muy alta en la fabricación de dos cámaras con lentes semejantes.
- **Dependencia de la ubicación:** Las aberraciones son más visibles en las regiones fronterizas de la imagen ya que la distorsión radial es una función del radio. Por tanto, cuanto más alejado está la línea del centro, más fuerte es la distorsión. Esto causa desplazamientos a los canales de color para hallar las posiciones de los píxeles.
- **Dependencia del contenido:** Es difícil estimar los parámetros de aberración en las imágenes con contenido de escena plana. Esto se debe a que en las distorsiones geométricas las imágenes con línea recta no son capaces de

estimar dichos parámetros de distorsión de manera correcta. Por otro lado, las aberraciones cromáticas se hacen visibles como franjas de color en áreas de alto contraste y en los bordes de los objetos.

- **Dependencia de configuración de la cámara:** Los valores de los parámetros estimados son diferentes en cada dispositivo debido a los ajustes de la cámara de cada usuario. Por ejemplo, para las imágenes capturadas con cámara específica y zoom manual diferente, el parámetro de distorsión de lente medido se extenderá en un amplio rango.
- **Coste computacional:** El coste temporal aumentara por los algoritmos aplicados y de la misma manera el coste computacional.

La distorsión radial de la lente (LRD, del inglés *Radial Distortion of the Lens*) es una de la más importante dentro de las distorsiones, especialmente cuando se usan lentes de gran angular de bajo coste. Choi et al. encontraron por primera vez que el nivel de LRD es diferente entre los distintos fabricantes de lentes, por lo que este valor se modifica con la variación de la distancia focal para la misma cámara[SCLW06]. En el trabajo se implementó un enfoque de optimización de características con un análisis discriminante por etapas para determinar las características específicas, y posteriormente identificar la fuente. Hwang et al [HPH14] propone la identificación de la fuente basada en la interpolación a través de la corrección de la distorsión de la lente, ya que, la interpolación para la corrección de la distorsión de la lente es específica del fabricante.

3.3. Técnicas Basada en la Aberración Cromática

La aberración cromática se produce cuando la misma longitud de onda no converge en la posición del plano focal. Se engloban en dos tipos la aberración: longitudinal y lateral. Las técnicas para la identificación de la fuente se basan en la aberración cromática lateral de las diferentes longitudes de ondas. Estas aberraciones producen distintos aumentos en el sensor y, por consiguiente,

modifica el tamaño de la imagen. Cada cámara contiene una lente diferente que experimenta varias aberraciones cromáticas. Estas aberraciones generan una desalineación entre los canales de color. Johson y Fraid [JF06] observaron las diferencias entre los canales de color con estimaciones de usos globales.

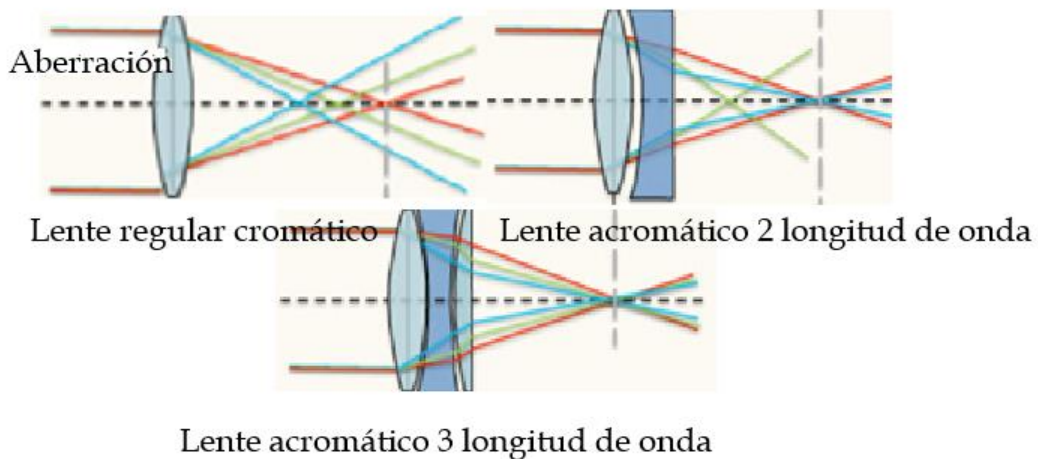


Figura 3.2: Ejemplo de aberración cromática.

3.4. Técnicas Basadas en las Imperfecciones del Sensor

Las técnicas basadas en las imperfecciones del sensor que dejan sobre las imágenes son las siguientes [GBK+01]: defectos del pixel, patrón del ruido del sensor (SPN, del inglés *Sensor Pater Noise*), ruidos de fotones y polvo del sensor.

3.4.1. Defectos del Pixel

En los defectos del pixel se examinan los pixeles calientes o los pixeles muertos para la identificación de la fuente de la imagen. En periodos de alta incidencia de luz se observa que ciertos pixeles tienen más brillo que otros, es decir, una celda del sensor se captura más la luz, este se satura y genera un pixel de color blanco, no obstante, en momentos de poca luminosidad esa celda de pixeles pueda volver a su estado normal, sin embargo, la posición de estos pixeles calientes seguirá siendo la misma. Los pixeles muertos, son defectos

físicos en el sensor, que en una determinada celda está apagada sin poder recibir luz, provocando la aparición de puntos negros en la imagen[AGRCSO+14].

En [GBK+01] se observó que los píxeles calientes y los píxeles muertos son utilizados para identificar la marca del móvil origen, sin embargo, en [VLCSK07] menciona que en las cámaras móviles de alta gama CCD, no encontraron ningún defecto visible en los píxeles además el uso de los algoritmos de post-procesamiento hacen que este método no es aplicable a todas las cámaras móviles.

3.4.2. Patrón del Ruido del Sensor

En [LFG06] se estudia el patrón de ruido de diferentes cámaras móviles determinando el ruido de patrón de referencia, identificando la marca del móvil específico que ha capturado la imagen, para la obtención de este patrón se estudió el promedio de ruido de múltiples imágenes utilizando un filtro de eliminación de ruido. En la identificación de la cámara a través de una imagen dada, se considera el patrón de referencia como una huella que cuya presencia en la imagen es establecida mediante el detector de correlación. Se señala que este método está afectado por algoritmos de procesamiento de la imagen tales como corrección gamma y la compresión JPEG.

En [CESR12] realiza un enfoque de atribución de la cámara origen tomando en cuenta escenarios abiertos, considerando que estos tienen acceso a las cámaras origen de las imágenes observadas. El planteamiento propuesto cuenta con tres fases: definición de las regiones de interés, descripción de las características y la atribución de la cámara origen en un escenario abierto.

En las regiones de interés se explica, que en las imágenes se pueden observar diferentes zonas que contengan información distinta sobre la marca del dispositivo móvil, tomando nueve regiones de la imagen que proporcionen

información importante, por lo general coinciden con el eje principal de la lente, debido a que algunas cámaras tienen efectos de bordes oscuros, causando una reducción de brillo en la imagen o saturación, en consecuencia, pueden tener más ruido.

Para determinar las características en [CESR12] calcula el SPN considerando los canales R, G, B, por separado. De igual forma el SPN considerando el canal Y (luminancia) que es una combinación de los canales de color R, G y B (como una versión en escala de grises de la imagen) generando 36 patrones de ruido de referencia para representar una imagen, las imágenes se clasifican en la fase de entrenamiento como positivas las observadas en la investigación y negativas las restantes del grupo. Los resultados de los experimentos mostraron una precisión del 94,45% con variaciones mínimas dependiendo de los patrones abiertos elegidos.

Técnica basada en	Características	Observaciones
Defectos del Pixel	Examina los pixeles calientes o los pixeles muertos para la identificación de la fuente de la imagen. Estos son defectos físicos en el sensor, que en una determinada celda está apagada sin poder recibir luz, provocando la aparición de puntos negros en la imagen.	Es poco eficiente en las cámaras móviles de alta gama CCD ya que no encuentra ningún defecto visible en los pixeles y usan los algoritmos de post-procesamiento.
Ruido del sensor	Realiza un promedio de ruido de múltiples imágenes utilizando un filtro de eliminación de ruido. Esta se basa en el análisis de wavelet. Se considera el patrón de referencia como una huella que cuya presencia en la imagen es establecida mediante el detector de correlación.	Se señala que este método está afectado por algoritmos de procesamiento de la imagen tales como corrección gamma y la compresión JPEG.
Aberración cromática	Basada en las diferentes longitudes de ondas, produciendo distintos aumentos en el sensor, lo que varía el tamaño de la imagen.	Cada modelo de cámara contiene una lente diferente que experimenta varias aberraciones cromáticas que generan una desalineación entre los canales de color.
Demosacing	Reconstruye la imagen haciendo uso de los datos recogidos por el sensor de imagen recubierto por el filtro de color.	No serviría de mucho sin un buen algoritmo de interpolación cromática.

Tabla 3.1: Técnicas de identificación de la fuente respecto al hardware.

3.5. Técnicas basadas en el Software del Dispositivo

Los artefactos relacionados con el software durante el proceso de formación de la imagen producen dos grupos diferentes de características de la imagen [JWWABA16]:

- Las características determinadas usando las estadísticas de la imagen.
- Las características determinadas por los coeficientes de los algoritmos de procesamiento como el algoritmo de interpolación cromática “demosaijing”, balanceo de blancos, corrección gamma, reducción de ruido, etc.

Estas características, conocidas como irregularidades de la imagen, contiene información que puede ser útiles en la identificación de la fuente o identificar diferentes condiciones de captura.

3.6. Características de la Imagen

En [JWWABA16] se introdujo las siguientes características para calcular los artefactos estadísticos intrínsecos dejados por diferentes cámaras: características de color, métricas de calidad de imagen (IQM, del inglés, *Image Quality Metrics*) y estadísticas del dominio wavelet. Se estudió las características básicas utilizando SVM (del inglés, *Support Vector Machine*) para identificar las marcas de la cámara de origen de las imágenes. En el estudio se hicieron varias modificaciones y se mejoró el rendimiento de la cámara basada en funciones estadísticas (modelo de atribución). El algoritmo de optimización GA (del inglés, *Genetic Algorithm*) comenzó con la generación aleatoria de la población de 10 matrices binarias. Para cada matriz binaria, los elementos de la matriz se indexaron de acuerdo con el índice de características en el vector, que consta de 33 características. El valor de “1” del elemento array, especifica que la función correspondiente se seleccionó en el vector de características, mientras que el valor de “0” se representa para los que no fueron seleccionados.

Este proceso fue repetido hasta que la tasa de detección llegó al 95% para un total de 20 características óptimas. En los casos que no se cumpliera la condición, la nueva matriz binaria se iba generando y se reemplazó con la antigua utilizando las tres siguientes operaciones GA: reproducción, cruzamiento y mutación.

Los investigadores utilizaron un subconjunto de imágenes y analizaron su extensión de los datos básicos con el clasificador SVM. El conjunto de características extendidas constaba de 6 características de color adicionales que se calculaban a partir de la imagen corregida con punto blanco y la diferencia entre la versión original y la versión corregida de puntos blancos de una imagen. Los experimentos de evaluación exhaustiva en este estudio alcanzaron resultados confiables de la atribución de la cámara con un total de 46 características.

Tsai et al en [TWLY12] trabajaron individualmente con el clasificador SVM generando el subconjunto de características propuesto. En el siguiente nivel, los subconjuntos de características se agregaron utilizando la fusión de decisión basada en el recuento. Finalmente, el análisis forense de las 20 características óptimas seleccionadas tiene las siguientes especificaciones: el verde era el color primario y características relacionadas con el color, ya que el modelo Bayer-pattern CFA con filas de filtros R-G y G-B es ampliamente utilizado en cámaras digitales, entre las IQMs se seleccionaron 5 características: error cuadrático medio, error absoluto, fidelidad de imagen, correlación cruzada normalizada y error de magnitud de fase espectral;

Utilizando cinco grupos diferentes de características estadísticas: Características del color (12 características), IQMs (12 características), WS (36 características), las imágenes de diferencia (30 características) y la predicción. Errores (12 funciones). Asimismo, aumentaron la base de datos de cámaras y probaron imágenes bajo técnicas alternativas de manipulación. Eventualmente,

el experimento mostró que la contribución de cada conjunto de características es diferente. Para mejores resultados se aumentó el tamaño de las características IQM a 40 mediante el cálculo de las métricas de cada color. Se presentan tres técnicas diferentes de fusión de decisiones: fusión del nivel de confianza, fusión de rango-nivel y fusión de nivel abstracto. En general, el método propuesto comparó el rendimiento aberración del conjunto características de fusión, contra la fusión antes de la selección de funciones para teléfonos móviles atribución de modelo de cámara.

Por último, utilizaron el clasificador de modelo Naïve Bayessiano [Ber], para evaluar la tasa de detección de imágenes bajo diferentes manipulaciones. Se centró en la influencia de la no-linealidad, que son distorsiones causadas por la cámara digital en la imagen de orden superior estadísticas más el impacto de las operaciones de procesamiento Dominio Wavelet. Así, el método empleó características estadísticas de imagen bi-coherencia El método utiliza el algoritmo SFFS y clasificador SVM para seleccionar las características óptimas del conjunto fusionado los experimentos el método propuesto contra el conjunto de características estadísticas básicas

3.7. Métricas para Cada Banda de Color de Varios Momentos Generalizados de Demosaicing Residual.

En [LLC+12] la técnica de fusión de decisión fue empleado para fusionar las decisiones de tres clasificadores SVM. Los clasificadores fueron capacitados con respecto a tres grupos de características y un puntaje de confianza para tomar una decisión. Adoptado a tres técnicas de fusión de decisiones (i) nivel de confianza de fusión, (ii) fusión de rango-nivel y (iii) fusión de nivel abstracto. El método comparó el rendimiento del conjunto características de fusión, contra la fusión antes de la selección de funciones para los teléfonos móviles de distintos modelos. El método empleado Naïve de información de clasificaciones, evaluó

la tasa de detección de imágenes bajo diferentes manipulaciones. Meng et al [AH09]. Se enfocaron en la influencia de la no-linealidad de distorsiones causadas por la cámara digital en la imagen de orden superior estadística, más el impacto de las operaciones de procesamiento (Wavelet). Así, el método empleó características estadísticas de la imagen con bi-coherencia y coeficientes Wavelet.

4. CONTRIBUCIÓN

En este trabajo se ha diseñado un algoritmo de identificación de la fuente de imágenes de dispositivos móviles que analiza los artefactos relacionados con el software del dispositivo que generó la imagen. El algoritmo realiza la extracción de la huella presente en las imágenes analizando los algoritmos de configuración CFA y las funciones de interpolación de color.

4.1. Regularidades de Procesamiento Basados en los Componentes del DPI

El método de identificación de la fuente de la cámara por medio de los artefactos de procesamiento, consiste en estudiar los algoritmos que usa el DPI para formar la imagen, asumiendo que existen algunas diferencias en los algoritmos que forman la imagen entre las diferentes cámaras.

Algunos autores consideran que la configuración CFA (elección de la matriz) y los algoritmos de interpolación cromática de cada dispositivo móvil, poseen características únicas que las hace diferentes al resto, lo que produce estructuras de correlación distinguibles en imágenes capturadas [JWWABA16].

En [BSMA05] Bayram et al, estudiaron las trazas que dejan los algoritmos de interpolación utilizando el algoritmo (EM) del inglés (*the expectation maximisation*) para la identificación de la marca de la cámara. Este algoritmo fue utilizado en partes lisas y no lisas por separado. Se empleó el clasificador SVM con dos conjuntos de características para la selección de funciones SFS (*Sequence of Filter Selection*), el conjunto de ponderación coeficientes obtenidos a partir de una imagen, y la localización del pico y magnitudes en el espectro de frecuencias.

La interpolación cromática o demosaicing introduce una correlación (dependiendo del algoritmo utilizado) entre los valores de los colores de los píxeles de la imagen. Se puede extraer estas correlaciones entre los píxeles de una imagen para diferenciar los algoritmos de interpolación y así poder saber la marca y el modelo desde la cual se tomó la imagen.

En [WTZ+15] se estudió un algoritmo no-intrusivo, el cuál es propuesto en [SWL07] para estimar los coeficientes de interpolación. Esta estimación consta de dos partes:

La primera los coeficientes de interpolación son estimados de manera preliminar con un modelo lineal. Los píxeles de la imagen son inicialmente divididos en tres categorías siguiendo la siguiente información de textura:

$$\textit{Horizontal: } \{Hx, y - Vx, y\} > T \quad (3.1)$$

$$\textit{Vertical : } \{Vx, y - Hx, y\} > T \quad (3.2)$$

$$\textit{Smooth: } \{Hx, y - Vx, y\} \leq T \quad (3.3)$$

Después de que estos coeficientes de interpolación hayan sido estimados, se obtiene un error de interpolación, calculado mediante una suma ponderada de errores de nueve categorías de píxeles, para evaluar la veracidad de la estimación. Además, estas estadísticas de detección deducidas por los errores se obtienen como un índice de clasificación para buscar diferentes patrones CFA. Debido a la alta complejidad se propone una simplificación del proceso de patrón CFA, usando un patrón típico Bayer diagonal para el CFA.

En [JWWABA16] se presenta un comparativo general entre las diferentes estrategias para superar los retos debido a la configuración CFA y a las características de interpolación de cada fabricante. [BSMA05] En este estudio se midió las huellas del algoritmo de interpolación de color usando el algoritmo EM para la identificación de la marca de la cámara origen. Las partes de las expectativas se centra en las variables desconocidas usando las estimaciones

actuales de los parámetros y condiciones sobre las observaciones, mientras que la parte de maximización predice una nueva estimación de los parámetros. Estos pasos se repiten hasta que ocurre la convergencia.

Eventualmente, el algoritmo EM genera dos salidas: un algoritmo de dos dimensiones y la estimación del coeficiente de ponderación que introducen el importe de la de cada píxel en el núcleo de interpolación.

El clasificador SVM empleó dos conjuntos de características para la selección de funciones SFFS: el conjunto de ponderación de los coeficientes obtenidos a partir de una imagen, y la localización del pico y magnitudes en el espectro de frecuencias.

El algoritmo de detección de interpolación propuesto se calculó derivado de segundo orden de cada fila y lo promedió en todas las filas. Proyectando la señal de pseudo-varianza en el dominio de la frecuencia, las ubicaciones de los picos de la señal de varianza revelan la velocidad de interpolación y la magnitud de los picos determina el método de interpolación. Además, emplearon la energía de cada componente de frecuencia con respecto a otra frecuencia componentes para todas las bandas de color.

Método	Características	Retos	Estrategias
[BSMA05] [BSMA06] [BSM06] [SBSM07] [BSM08]	Coeficiente de ponderación, localización de picos y magnitud de la frecuencia. Espectro de regiones lisas y no lisas utilizando el algoritmo EM.	La información no prioritaria se asume en el tamaño del núcleo de información, los coeficientes DC introducen picos en la derivada de segundo orden, lo que induce la operación de interpolación Revela características lineales en la parte lisa de la imagen Grandes tasas de error alcanzadas con SPN básico	Ejecución de varios experimentos para estimar los tamaños del núcleo Ignora los picos debido a la compresión JPEG
[[CAS07] [HF10] [JHF12]	PCA de la matriz de coeficientes de ponderación a partir del modelo de correlación cuadrática entre píxeles.	Grandes dimensiones, las características del algoritmo de interpolación tienen que seguir la agrupación original.	Se ha empleado el PCA para reducir la dimensión de la matriz de coeficientes de correlación cuadráticos entre píxeles [CAS07] utilizando tres capas de avance BP NN [HF10] [JHF12] proporcionando la solución analítica de tres capas de aprendizaje mediante el uso de CPPNs.
[SWL07]	Presentan coeficientes de ponderación del filtro 7 x 7 en cada tipo de región (regiones de gradiente horizontal y vertical) y canal de color.	Costo de cómputo debido al espacio de búsqueda. Dependencia de contenido, dependencia de ubicación y efectos de las operaciones de postprocesamiento.	Detección del patrón CFA. Imagen clasificada en tres regiones de textura. En el estudio estimaron los coeficientes de interpolación usando SVD para tratar con posibles valores de píxel ruidosos. Bloques seleccionados con valores de gradiente más altos
[MSGW08]	Ponderación (441) Estadísticas de ruido (60)	Caracterización local de los coeficientes de interpolación de color	Se propone una fusión de un conjunto de dos características diferentes de procesamiento de señal para incrementar la robustez del identificador de cámara.
[CK08]	Coeficientes de ponderación de la derivada de 2° orden (y superior) del modelo de correlación de la imagen.	Costo de cómputo debido al espacio de búsqueda. Dependencia de contenido, dependencia de ubicación y efectos de las operaciones de postprocesamiento.	Suponiendo un filtro Bayer CFA se agruparon las muestras de demosaicing en tres grupos basándose en el patrón de vecindad para Bayer CFA (es decir, X, Y, OXY y U, V, OUV) Se eliminaron las influencias de los componentes de DC desiguales en los tres canales de color para detectar algoritmos de interpolación de color usando correlaciones entre canales.
[CK09a] [CK09b] [CK10]	<ul style="list-style-type: none"> • Coeficientes de ponderación (1248) • Estadísticas de error (256) 	Las características del algoritmo de interpolación tienen que seguir el agrupamiento original	Se obtuvieron 16 grupos de muestras de demosaicing utilizando para el patrón de vecindario del filtro de Bayer CFA de dos niveles EMRC para descubrir con precisión la

	<ul style="list-style-type: none"> Tamaños de grupos normalizados (32) 		<p>agrupación original. Conjunto compacto derivado de ERE de características de Eligen discriminantes.</p>
[WKY09]	Coeficientes de ponderación (1022)	<ul style="list-style-type: none"> Imágenes de cámaras desconocidas. Efectos de las operaciones de doble compresión JPEG. 	<ul style="list-style-type: none"> Clasificó outliers mediante la adaptación de SVM de una clase. Impacto en la matriz de covarianza reducida.
[CCL11]	Matrices de conteo de valores intermedios	<p>Costo computacional debido al espacio de búsqueda El recorte aumenta la probabilidad de que el bloque contenga una región de textura o un borde, y hace confuso el método propuesto. La información sobre el patrón CFA se dañó gravemente con la compresión JPEG con pérdidas</p>	<p>Estimación del patrón CFA La robustez de las características propuestas debería estudiarse en el futuro.</p>

Tabla 3.2: Comparación de las técnicas de identificación de la fuente basadas en la configuración CFA y las funciones de interpolación de color.

4.2. Conceptos Generales

A continuación, se definen brevemente las técnicas en las que se basa el algoritmo, para entender su funcionamiento.

4.2.1. Esperanza Maximización

El algoritmo EM consta de dos pasos resaltantes a la hora de su implementación, la primera fase es la expectativa, seguido de maximización de sus datos basándose en la agrupación por diferencias, en esta investigación se propone separar los grupos entre los píxeles que estén o no correlacionados con sus vecinos, de esta manera poder hallar la matriz de kernel utilizada en las diferentes marca de móviles, obteniendo distintos valores para cada matriz de kernel indicando la diferencia de los algoritmos implementados en la formación de la imagen. [BSMA05]

En el algoritmo el paso de la expectativa es con respecto a las variables subyacentes desconocidas, en nuestro caso las variables de la matriz utilizada para la obtención del kernel específico con valores aleatorios, para ello se da uso a la estimación actual de los parámetros, y a la condicionada de las observaciones. El paso de maximización proporciona una nueva estimación de los parámetros. Estos dos pasos se iteran hasta la convergencia, generando dos salidas. Una es la matriz del kernel, llamada mapa de probabilidad, con cada entrada que indica la similitud de cada píxel de imagen con uno de los dos grupos de muestras, es decir, los que están correlacionados con sus vecinos y los que no. En este mapa las regiones identificadas por la presencia de patrones periódicos indican la parte de la imagen que ha sido sometida a una operación de muestreo, la otra salida producida es la estimación de los coeficientes de ponderación que designan la contribución de cada píxel en el núcleo de interpolación. Se puede observar en la figura 5.1 el esquema del procedimiento del algoritmo.

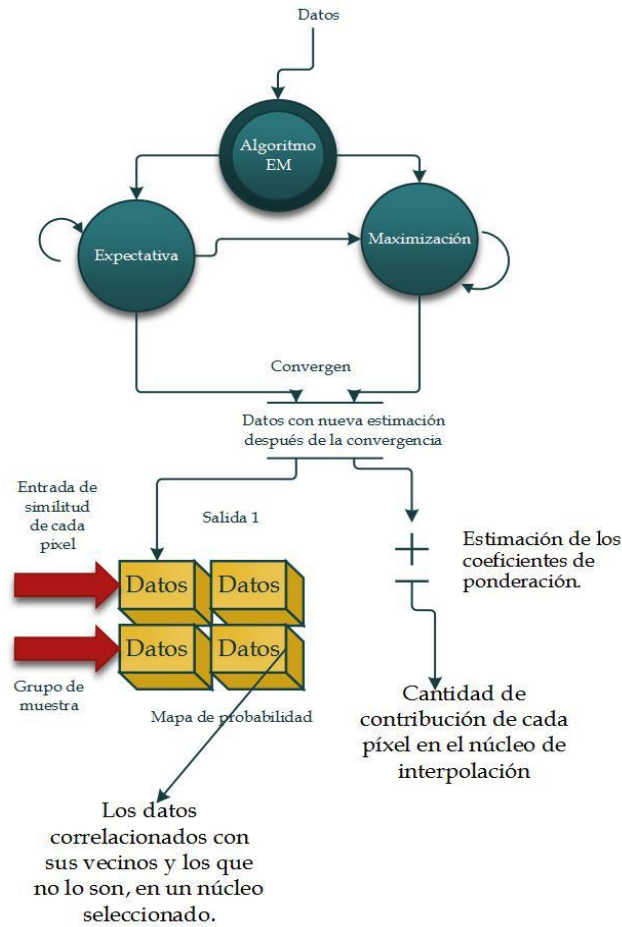


Figura 4.1: Técnica del algoritmo EM.

4.2.2. Detección de Interpolación

La interpolación es el proceso de estimar el valor de una señal en las posiciones intermedias a las muestras originales. Generalmente, esto se logra ajustando una función continua a las muestras conocidas y evaluando la función. Para evitar el aliasing sin degradación de la señal válida, debe usarse un filtro de paso bajo ideal. Este algoritmo tiene como objetivo determinar si se ha producido una interpolación de orden inferior, es decir una interpolación bilineal o bicubica.[Gal05]

El algoritmo funciona calculando la segunda derivada de cada fila de la imagen. Seguido de esto, los valores absolutos de cada fila de su segunda derivada, estos se promedian juntos para obtener una media.

Esta señal es proporcional a la señal de varianza, si una imagen ha sido interpolada, este rastro mostrará una periodicidad relacionada con la tasa de interpolación. Finalmente, se implementa la discreta de Fourier de la media de la segunda derivada para los picos y las frecuencias correspondientes de la localización de los picos que se utilizan para determinar la frecuencia del periodo. Con esto poder identificar qué tipo de frecuencia presenta el sensor en la formación de la imagen como en la Figura 5.4

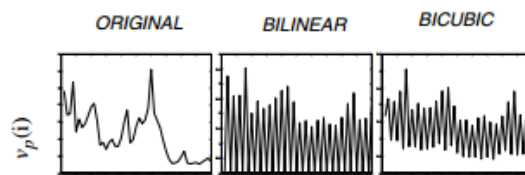


Figura 4.2 Tipo de frecuencia de la segunda derivada de una imagen interpolada.

4.2.3. Estrategias de Selección de Características.

El algoritmo de selección de características engloba un conjunto de algoritmos de búsqueda exhaustiva que se usan para reducir un espacio característico inicial a un subespacio el cual, este debe tener menor dimensión que el inicial. El objetivo es seleccionar un subconjunto de características que sean más relevante para el problema dado reduciendo el error de generalización del modelo, eliminando características irrelevantes, de esa manera se simplifica la complejidad del clasificador SVM y mejora la precisión de clasificación.

Las características seleccionadas son óptimas según la función de criterio elegida para guiar el algoritmo. Sin embargo, el coste computacional hace a menudo este acercamiento inadecuado, así requiriendo algoritmos alternativos, en nuestro caso, la unión entre el algoritmo EM con el SFFS.

En este trabajo hemos implementado dos enfoques sub-óptima con una solución única, también conocido como envolturas [4]. En el primero, el

subconjunto seleccionado empieza vacío y las características se insertan optimizando una función de criterio hasta que se cumple una condición de parada, que se basa en el tamaño del subconjunto, comparando con las características dentro del subconjunto. Este método se conoce como *Sequential Forward Selection* (SFS) [5]. Aunque simple y eficiente, SFS presenta un inconveniente indeseable conocido como efecto de anidamiento. Este efecto surge porque las características seleccionadas nunca se descartan.

Se realiza una técnica simple que mide la discriminación de dos conjuntos de números reales. Dado los vectores de entrenamiento, se define como:

$$F(i) \equiv \frac{(xi^{(+)} - xi)^2 + (xi^{(-)} - xi)^2}{\frac{1}{n_+ - 1} \sum_{k=1}^{n_+} (x_{k,i}^{(+)} - xi^{(+)})^2 + \frac{1}{n_- - 1} \sum_{k=1}^{n_-} (x_{k,i}^{(-)} - xi^{(-)})^2}$$

Las X_i indica la media de la característica del conjunto positivo y negativo el numerador indica la discriminación entre los conjuntos positivo y negativo, el denominador indica el número dentro de cada uno de los dos conjuntos. Cuanto mayor sea la puntuación F , más probable es que esta característica sea más discriminatoria. Por lo tanto, utilizamos este puntaje como un criterio de selección de características. Una desventaja de F -score es que no revela información mutua entre las características. A pesar de esta desventaja el f -score es bastante efectivo. El patrón que se sigue en el algoritmo se muestra en la Figura 4.3.

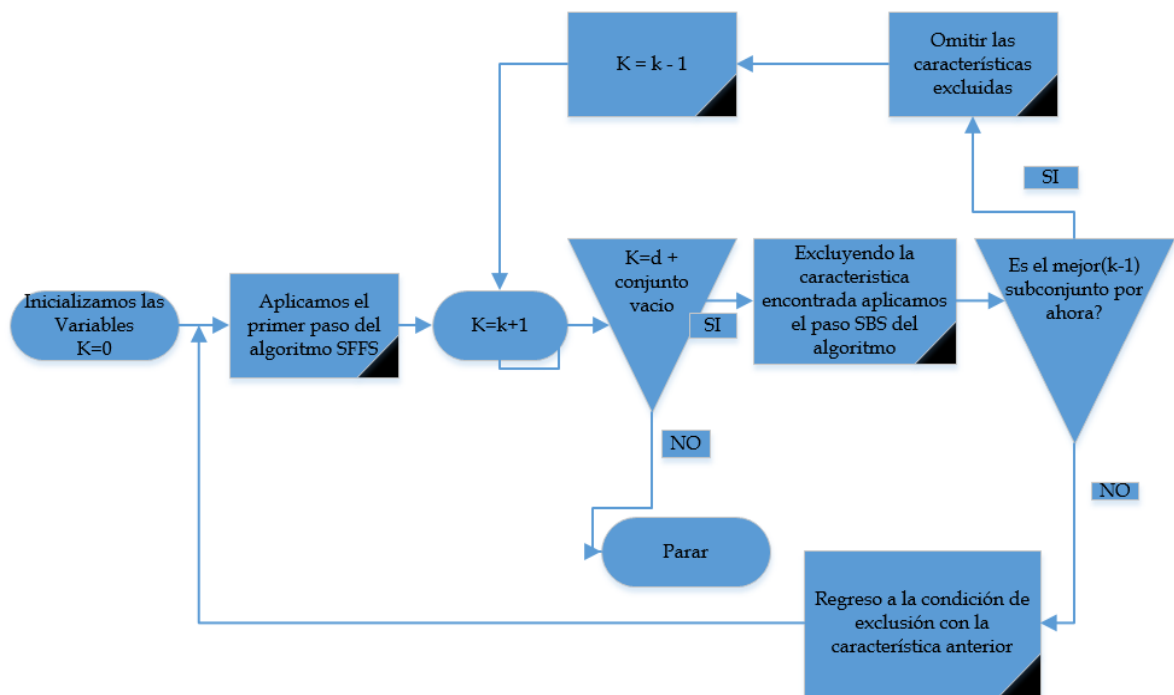


Figura 4.3: Esquema del algoritmo SFFS.

En nuestra investigación seleccionamos características con altas puntuaciones F y luego aplicamos SVM para entrenamiento y la predicción. El procedimiento se puede observar en la Figura 4.3.

Para conjuntos de datos en esta competencia, existe una brecha bastante clara entre las puntuaciones altas y las inferiores. Podemos automatizar este paso, por ejemplo, añadiendo gradualmente rasgos de puntuación alta, hasta que la precisión de validación disminuya.

4.3. Especificación del Algoritmo

El algoritmo de identificación de la fuente de imágenes digitales propuesto utiliza la técnica de Esperanza-Maximización (EM) para extraer los pesos de los coeficientes de ponderación que designan la probabilidad de contribución de cada píxel en el núcleo de interpolación. La matriz del núcleo utilizado en la formación de la imagen es estimada a través de un conjunto de N etiquetas

asociadas a cada una de las observaciones deseadas.

Una vez extraídas las características de un conjunto de imágenes se utiliza un sistema de aprendizaje supervisado para relacionar las imágenes investigadas con el dispositivo móvil que la generó. Este sistema consta de dos fases: entrenamiento y predicción.

Para mejorar los resultados de clasificación se ha propuesto una pre-selección de las características que aporten más información diferenciadora y mejore la tasa de acierto. Para ello, se utiliza la técnica de selección de características SFFS. El principal objetivo de SFFS es la selección del mejor subconjunto de características del conjunto total basándose en la ponderación de los coeficientes obtenidos. Se ha utilizado como apoyo la librería `fselect.py` de la implementación LIBSVM en Python [lib] se ha adaptado a las necesidades de este trabajo.

4.3.1. Extracción de los Pesos de los Coeficientes de Interpolación

El algoritmo implementado tiene como objetivo estimar simultáneamente un conjunto de muestras periódicas que están correlacionadas con sus vecinos, y la forma específica de estas correlaciones. Se asume que cada muestra pertenece a uno de dos modelos. El primer modelo, M1, corresponde a aquellas muestras y_i que están correlacionadas con sus vecinos. M1 se puede estimar mediante la ecuación (4.1)

$$M1: y_i = \sum_{k=-N}^N \alpha_k y_{i+k} + n(i) \quad (4.1)$$

donde, N es el tamaño del matriz núcleo, α son los valores del conjunto del matriz núcleo que al comienzo del algoritmo son aleatorios con un rango normalizado, $y(i+k)$ es la matriz imagen, (n) denotan de forma independiente, e idénticamente distribuida, el primer modelo M1 son las muestras extraídas de

una distribución Gaussiana con media cero y varianza desconocida

El segundo modelo M2 corresponde a los pixeles que no están correlacionados, es decir, los pixeles que se generan mediante un proceso de valores atípicos.

El algoritmo EM es un algoritmo iterativo de dos pasos: El primer paso corresponde con la esperanza (E) que se estima la probabilidad de que cada muestra pertenezca a cada modelo, el segundo paso corresponde al paso de maximización (M) que estima la forma específica de las correlaciones entre muestras.

En el paso E, la probabilidad de cada muestra y_i pertenecientes al modelo M1 se pueden obtener utilizando la regla de Bayes mostrada en la ecuación (4.2).

$$\Pr\{y_i \in M1|y_i\} = \frac{\Pr\{y_i|y_i \in M1\} \Pr\{y_i \in M1\}}{\sum_{k=1}^2 \Pr\{y_i|y_i \in M1\} \Pr\{y_i \in M1\}} \quad (4.2)$$

donde $\Pr\{y_i \in M1\}$ y $\Pr\{y_i \in M2\}$ se asume que son iguales a $1/2$, dado así que la probabilidad de que ese pixel haya sido generado por M1 se expresa como se muestra en la ecuación (4.3).

$$\Pr\{y_i \in M1|y_i\} = \frac{1}{\sigma\sqrt{2\pi}} \left[\frac{-(y_i - \sum_{k=-N}^N \alpha_x y_{i+k})^2}{2\sigma^2} \right] \quad (4.3)$$

donde, la varianza σ es de 0.075 inicialmente, α son los valores del conjunto de la matriz núcleo, $y(i+k)$ es la matriz imagen. La probabilidad de pertenecer al M1 está distribuido en el rango de valores posibles de Y_i , de la varianza al cuadrado. La fase (E) requiere una estimación de α , en la primera iteración se elige al azar.

En el paso M se estima una nueva varianza para α utilizando mínimos cuadrados ponderados y los pesos $w(i) \equiv \Pr\{y_i \in M1 | Y_i\}$, y $\alpha_0 = 0$. Esta función de error se minimiza calculando el gradiente con respecto a α , y

poniendo el resultado igual a cero, para resolver α se calcula con la ecuación (4.4).

$$\vec{\alpha} = (Y^T W Y)^{-1} Y^T W \vec{y} \quad (4.4)$$

donde W es la matriz de pesos con la diagonal $w_i[i][i]$, el paso E y el paso M se ejecutan iterativamente hasta que se alcanza una estimación estable de α .

La implementación consta de:

- Entrada: Matriz de la imagen.
- Salida: Estimación de los coeficientes de ponderación y la matriz del mapa de probabilidad.

Los Algoritmos 1 y 2 presentan el pseudocódigo de los pasos de Esperanza y Maximización. La Fase de Esperanza se repite hasta que α converja.

Algoritmo 1: Fase de Esperanza

1. Inicializar una matriz $\{a(0)_{u,v}\}$ con valores pseudoaleatorios de tamaño $N \times N$ ($N=5$)
2. Establecer p_0 del rango de valores posibles de $f(x, y)$ en nuestro caso la imagen.
3. $n=0$
4. Mientras no converge y n sea menor que el tamaño de la matriz núcleo:

a. for each de la imagen $f(x, y)$:

b. Calcular el error residual con:

$$r(x, y) = |f(x, y) - \sum_{u,v=-N}^N \alpha_{u,v}^n f(x+u, y+v)|$$

c. Calcula la condición de probabilidad y la probabilidad posterior con:

$$P(x, y) = \frac{1}{\alpha_{n\sqrt{2\pi}}} \exp[-r^2(x, y) / 2\alpha_n^2]$$

$$w(x, y) = \frac{P(x, y)}{P(x, y) + P_0}$$

d. Calcular los mínimos cuadrados con la ecuación (5.6) para $\{\alpha_{u,v}^{(n+1)}\}$

e. Realizar la nueva estimación de la varianza con

$$\sigma_{n+1} = \left(\frac{\sum_{u,v} w(x, y) r^2(x, y)}{\sum_{x,y} w(x, y)} \right)^{1/2}$$

f. $n=n+1$

4.3.2. Detección del Tipo de Interpolación

A continuación, se muestra el algoritmo estimar el valor de una señal en las posiciones intermedias a las muestras originales. Se puede estimar con la ecuación (4.5)

$$Vp(j) = \sum_{i=0}^R |Sp(i,j)| \quad (4.5)$$

donde, $Sp(i,j)$ es una señal bidimensional hecha de las señales de la segunda derivada para cada fila de la imagen. Las magnitudes de las filas de señales de segunda derivada se promedian juntas para formar una señal de pseudo-varianza. La señal $sp(i)$ contiene la información de frecuencia en $C - 2$ puntos, representando frecuencias normalizadas de 0 a $1 - \frac{1}{C-2}$ correspondiendo un factor de interpolación de:

$$\bar{N} = \frac{1}{F} \quad (4.6)$$

donde F es la Frecuencia calculada.

La DFT (en inglés, *Discrete Fourier Transform*) de una señal de tiempo o espacio, es la representación de la frecuencia de señal que puede determinarse mediante la detección de pico en la señal DFT. Los casos de interpolación bicúbica y bilineal tienen un pico pronunciado en la DFT, los picos se producen a la frecuencia normalizada de $1/2$, con un período correspondiente de 2.

Algoritmo 3: Algoritmo de Detección de Interpolación

1. Computar la segunda derivada de cada fila con:

$$s p(i, j) = 2 * p(i, j) - p(i, j + 1) + p(i, j - 1)$$

2. Hallar las medias de cada fila con respecto al paso 1.

$$Vp(j) = \sum_{i=0}^R |sp(i,j)|$$

3. Computar la transformada discreta de Fourier para hallar la frecuencia.

4. Determinar el factor de interpolación.

Los pasos anteriores se implementaron con bibliotecas alternativas de Python para su debido funcionamiento.

4.3.3. Selección de Características

El algoritmo de selección de características propuesto tiene como entrada el vector de características extraídas y el resultado del mismo es un vector con las características más relevantes y los parámetros óptimos para la máquina de soporte vectorial.

El funcionamiento del algoritmo es el siguiente:

Primero, se elige un umbral de elección de características altas y bajas como se ha explicado en el apartado 4.1.2. A continuación se buscan las características que tengan mayor representación en el conjunto para agregarlas a un conjunto vacío con las características más relevantes.

Para cada umbral se escogen las características más relevantes del subconjunto y se almacenan en un vector con las características seleccionadas con su respectivo índice. Las características de disminución con F-Score deben estar por debajo de este umbral. Posteriormente, se dividen aleatoriamente los datos de entrenamiento y se almacenan los nuevos datos de entrenamiento. Esto se repite cinco veces y luego se calcula el valor de medio del error.

A continuación, se realiza la elección del umbral con el valor de medio del error más bajo y se calculan las características de disminución que están por debajo del umbral seleccionado.

Finalmente, se realiza las fases de inclusión y exclusión de las mejores características obtenidas del subconjunto.

Algoritmo 4: Fase de inclusión de características

1. Se crea un conjunto vacío X
2. Mientras no termine de buscar
 - a. Se realiza una búsqueda de las características más relevantes del conjunto X_k de entradas:

$$X^+ = \underset{c \in (Y - X_k)}{\operatorname{argmax}} J(X_k)$$

- b. Se agrega al conjunto esa característica relevante.

$$X_{K+1} = X_K + X^+$$

- c. Se obtiene la mejor característica del subconjunto.

$$J_{max} = \max(J(X_{K+1}), J_{max})$$

Algoritmo 4.1: Fase de exclusión de características

- d. Buscar la última característica insertada

$$X^- = \underset{c \in (X_k)}{\operatorname{argmax}} J(X_k)$$

- e. Juzgar si pertenece a las mejores características obtenidas del subconjunto:

- f. Si $\text{If} (J(X_k) - X^-) > J_{max}$

- g. Entonces

$$J_{max} = J(X_k - X^-)$$

- h. Exclusión de característica

$$X_{k-1} = X_k - X^-$$

$$k = k-1$$

- i. Volver al paso de Exclusión

- j. Else, volver al paso de Inclusión
-

4.4. Herramientas de Desarrollo

Para el desarrollo de este trabajo se utilizaron las siguientes herramientas:

- **Python:** Se ha utilizado el lenguaje de programación Python en su versión 2.7.6 ya que tiene una licencia de código abierto (*Python Software Foundation License*). Esta licencia es compatible con la Licencia Pública General de GNU a partir de la versión 2.1.1
- **Mongo:** Se ha utilizado la base de datos NoSQL programada en MongoDB.

5. EXPERIMENTOS Y RESULTADOS

Para evaluar la eficacia de los algoritmos propuestos y los resultados obtenidos se realizaron 3 tipos experimentos: Análisis de fotografías de dispositivos móviles de un mismo fabricante y diferentes modelos, detección de tipos de interpolación, y evaluación de los pesos de los coeficientes de interpolación. Los experimentos se realizaron con diferentes resoluciones y combinaciones de características. Los resultados de los experimentos muestran que el algoritmo de selección de características mejora la tasa de acierto. En la Tabla 5.1 se presenta un resumen de los experimentos realizados.

Experimentos	Recorte	Marca	Modelos	Número de modelos distintos
Una marca con distintos modelos	256 px 512px 1024px	LG	e400	1
			Nexus 5	4
			p760	1
		IPHONE	4	1
			4S	2
			5	2
			6 Plus	2
			7	1
		SAMSUNG	A5_16	1
			Galaxy Nexus	1
			Galaxy S4 mini	1
			Galaxy S3	1
			Galaxy S5	1
			Galaxy S6	1
			Galaxy S Plus	1
Galaxy S	1			
Galaxy Ace s5830	1			
Galaxy Ace s5830m	1			
Galaxy Grand 2	1			
Una marca con dos modelos distintos	256 px 512px 1024px		6 plus	2
			4S	2
			e400	1
			Nexus 5	1

Tabla 5.1: Distintos Experimentos Implementados.

5.1. Análisis de Móviles del mismo Fabricante y Diferentes Modelos

En este experimento se probaron 6 dispositivos de la misma marca con diferentes modelos, el nexus presentaba cinco dispositivos de la misma marca y modelo, el e400 un solo dispositivo al igual que el e760.

Las fotografías obtenidas son de escenas diferentes sin ninguna restricción. En este experimento se deseaba comprobar la eficiencia del algoritmo de selección de características con respecto a los antiguos algoritmos implementados. La Tabla 5.1 apartado A resume los principales parámetros utilizados.

Algoritmo en Conjunto	Porcentaje de Acierto Sin SFFS	Porcentaje de Acierto Con SFFS
IQM	27,5%	58,833%
COLOR	50%	49%
WAVELETS	50,66%	41%
TODAS	50%	51,33%

Tabla 5.2 Resultados Experimento P1_A

Los resultados obtenidos en la identificación de la fuente se muestran en la tabla 5.2. El número de fotografías de entrenamiento fueron de 50 cada móvil en total unas 600 fotos de entrenamiento y otras 600 para pruebas. Dado los resultados del experimento se puede concluir que el algoritmo de selección de característica mejora en los casos de los algoritmos que generan más de 10 características en la imagen.

En la Tabla 5.2 se observan los valores de porcentajes de entrenamiento con el algoritmo IQM en conjunto con el algoritmo de selección de características, respectivamente, se muestra un porcentaje alto de características conseguidas en el algoritmo de un 58.833%.

En el siguiente experimento se analizaron 8 dispositivos de la misma marca con diferentes modelos.

Las fotografías obtenidas son de escenas diferentes sin ninguna restricción. En este experimento se deseaba comprobar la eficiencia del algoritmo de selección de características con respecto a los antiguos algoritmos implementados.

Algoritmo en Conjunto	Porcentaje de Acierto Sin SFFS	Porcentaje de Acierto Con SFFS	Recorte
IQM,Color, Wavelets,PRNU	50,83%	55,87%	256
IQM,Color, Wavelets,PRNU	57,337%	62,406%	512

Tabla 5.3 Resultados Experimento con 8 dispositivos

Los resultados obtenidos en la identificación de la fuente se muestran en la Tabla 5.3. El número de fotografías de entrenamiento fueron de 50 cada móvil en total unas 400 fotos de entrenamiento y otras 400 para pruebas. Dado los resultados del experimento se puede concluir que el algoritmo de selección de característica mejora, donde se puede apreciar que es posible la identificación entre modelos de la misma marca y modelo, también es notorio el nivel de recorte utilizando, cuanto más recorte utilizado mayor será el porcentaje de acierto.

En el siguiente experimento se analizaron 11 dispositivos de la misma marca con diferentes modelos. Las fotografías obtenidas son de escenas diferentes sin ninguna restricción. En este experimento se deseaba comprobar la eficiencia del algoritmo de selección de características con respecto a los antiguos algoritmos implementados.

Algoritmo en Conjunto	Porcentaje de Acierto sin SFFS	Porcentaje de Acierto con SFFS	Recorte
IQM,Color, Wavelets,PRNU	100%	100%	256
IQM,Color, Wavelets,PRNU	100%	100%	512
IQM,Color, Wavelets,PRNU	100%	100%	1024

Tabla 5.4 Resultados Experimento con 11 dispositivos

Los resultados obtenidos en la identificación de la fuente se muestran en la tabla 5.4. El número de fotografías de entrenamiento fueron de 100 cada móvil en total unas 1100 fotos de entrenamiento y otras 1100 para pruebas. Dado los resultados del experimento se puede concluir puede obtenerse un porcentaje de acierto exacto cuando no implica observar los dispositivos de la misma marca y modelo.

En el siguiente experimento se analizaron 4 dispositivos de la misma marca con diferentes modelos. Las fotografías obtenidas son de escenas diferentes sin ninguna restricción. En este experimento se deseaba comprobar la eficiencia del algoritmo de selección de características con respecto a los antiguos algoritmos implementados.

Algoritmo en Conjunto	Porcentaje de Acierto sin SFFS	Porcentaje de Acierto con SFFS	Recorte
IQM,Color, Wavelets,PRNU	55.8897%	62.4%	512
IQM,Color, Wavelets,PRNU	57%	67%	1024

Tabla 5.5 Resultados Experimento 4 dispositivos

5.2. Experimento de Detección de Tipos de Interpolación

En este experimento se probó 7 dispositivos de diferentes modelos, utilizando el algoritmo de interpolación bicubica o bilineal, para detectar las zonas suaves de la imagen, es decir, si se ha empleado zoom en cada foto, con ello poder obtener el tipo de frecuencia del sensor utilizado en las diferentes cámaras digitales, aportado en las gráficas del algoritmo.

Dispositivo	Foto sin Zoom	Foto con Zoom medio	Foto con Todo el Zoom
IPhone 7	Bilinear	Bilinear	Bilinear
Samsung A5 16	Bicubica	Bicubica	Bicubica
Samsung GSMJ500	Bicubica	Bicubica	Bicubica
Samsung GSMJ500	Bicubica	Bicubica	Bicubica
LENOVO	Bilineal	Bilineal	Bilineal
SAMSUNG S	Bicubica	Bicubica	Bicubica
SAMSUNG TAB2	Bicubica	Bicubica	Bicubica

Tabla 5.6 Resultados Experimento Zonas Suaves.

Los resultados obtenidos en la identificación de la fuente se muestran en la tabla 5.6. El número de fotografías de entrenamiento fueron de 30 cada móvil. Dado los resultados del experimento se puede concluir puede obtenerse un el tipo de frecuencia del sensor utilizado, con ello agregar una característica relevante en la identificación de la fuente de la imagen, en la implementación propuesta solo se podrá observar la frecuencia con las gráficas del algoritmo.

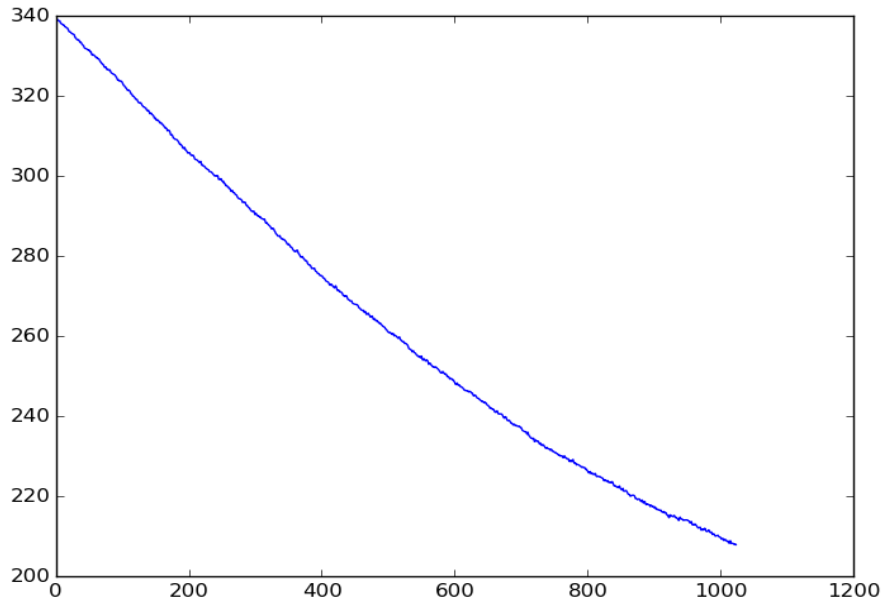


Figura 5.1: Ejemplo Samsung J500 sin Zoom

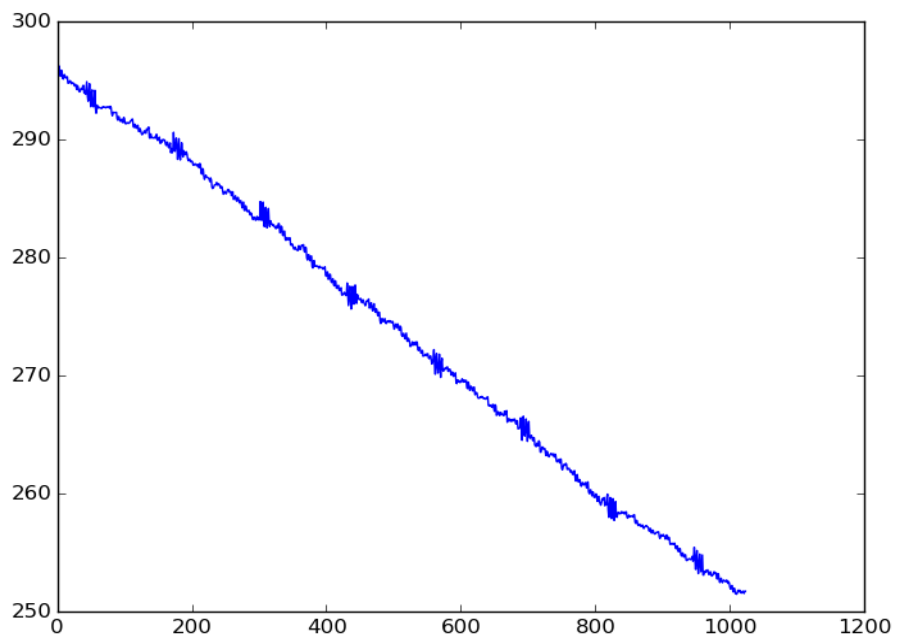


Figura 5.2: Ejemplo Samsung J500 con Zoom

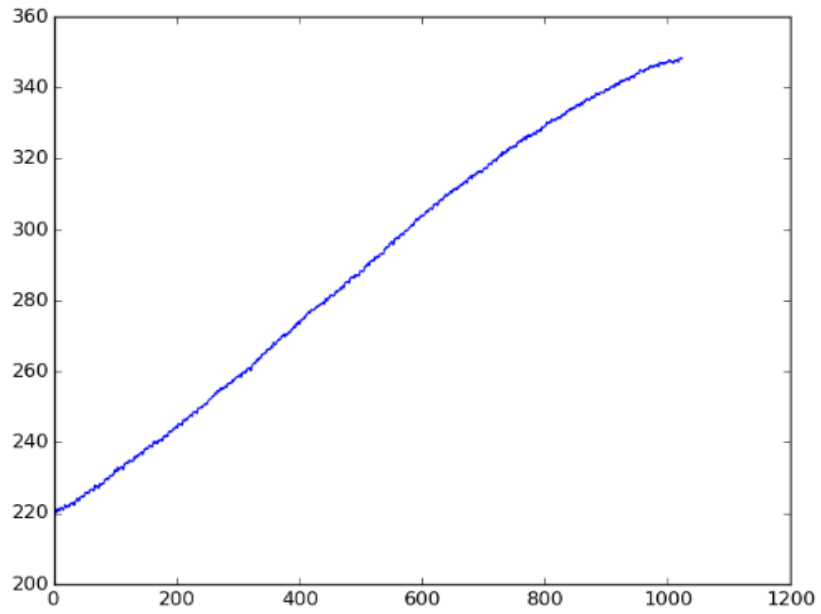


Figura 5.3: Ejemplo Lenovo sin Zoom

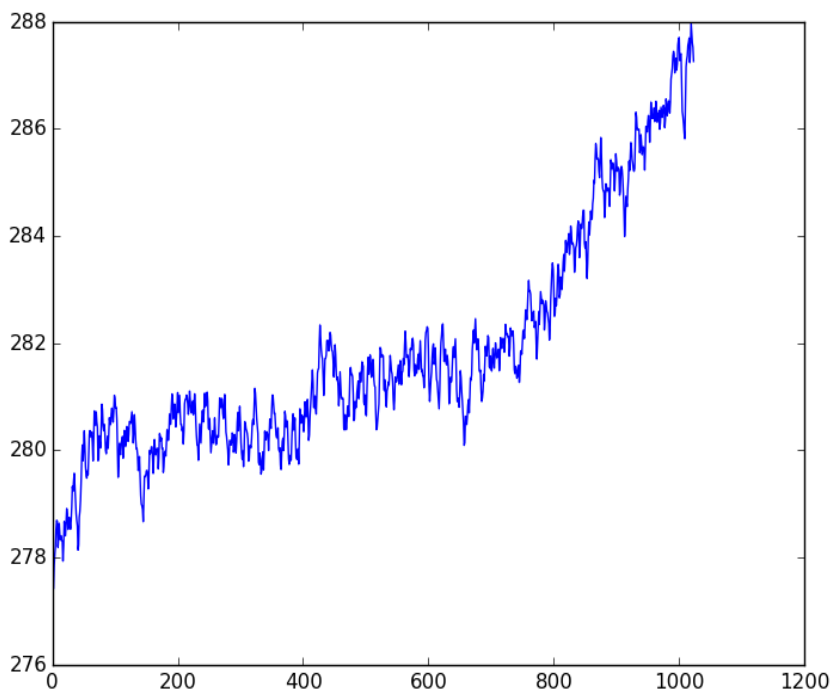


Figura 5.4: Ejemplo Lenovo con Zoom

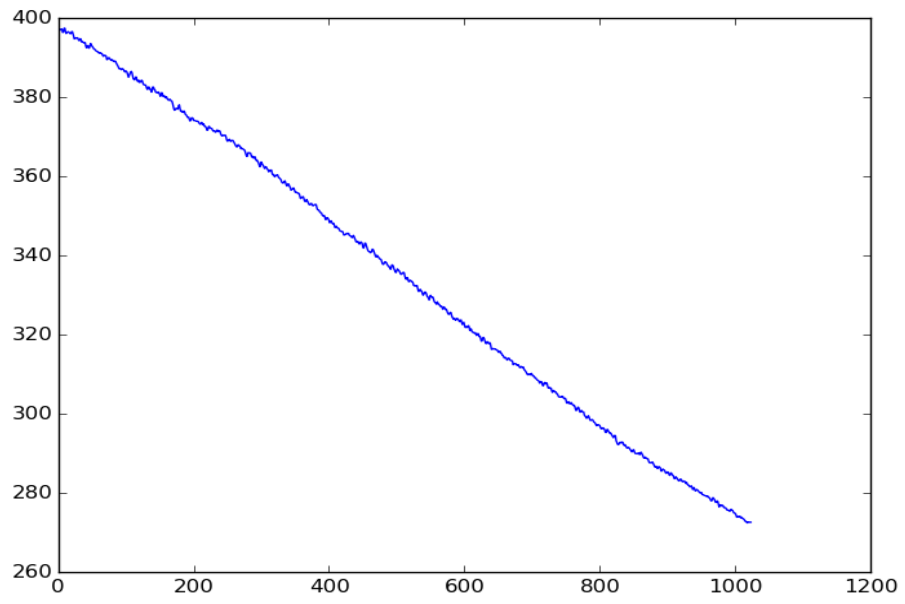


Figura 5.5: Ejemplo Iphone 6 sin Zoom

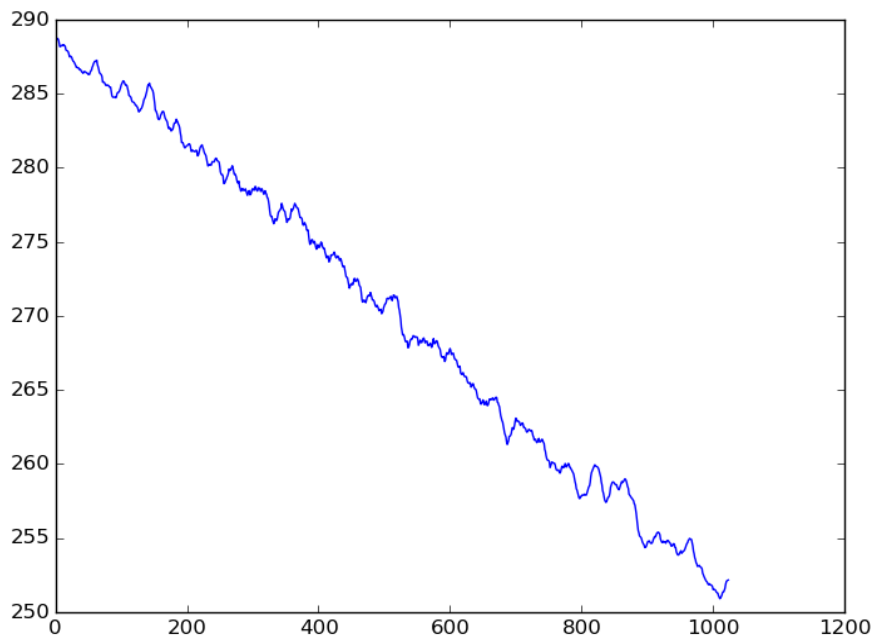


Figura 5.6: Ejemplo Iphone 6 con Zoom

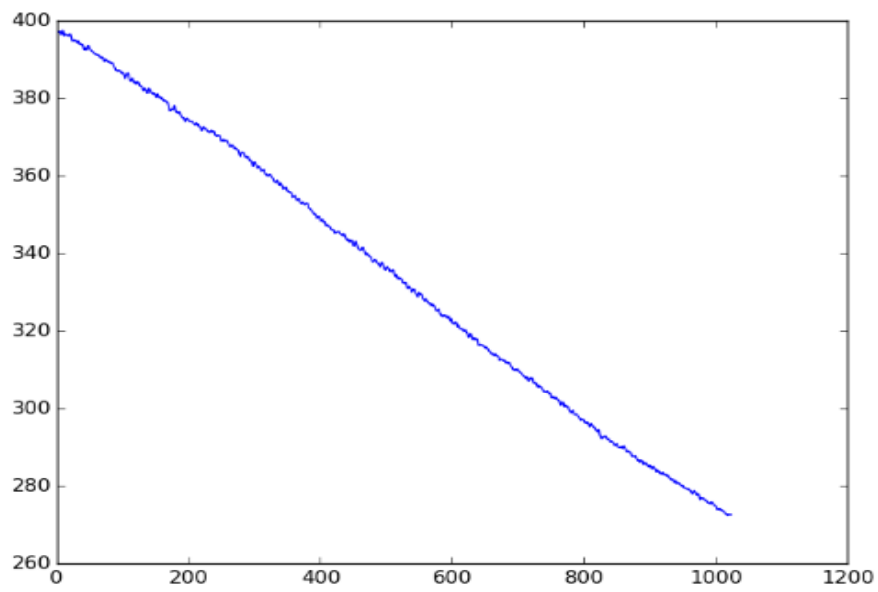


Figura 5.7: Ejemplo Samsung Tab 2 sin Zoom

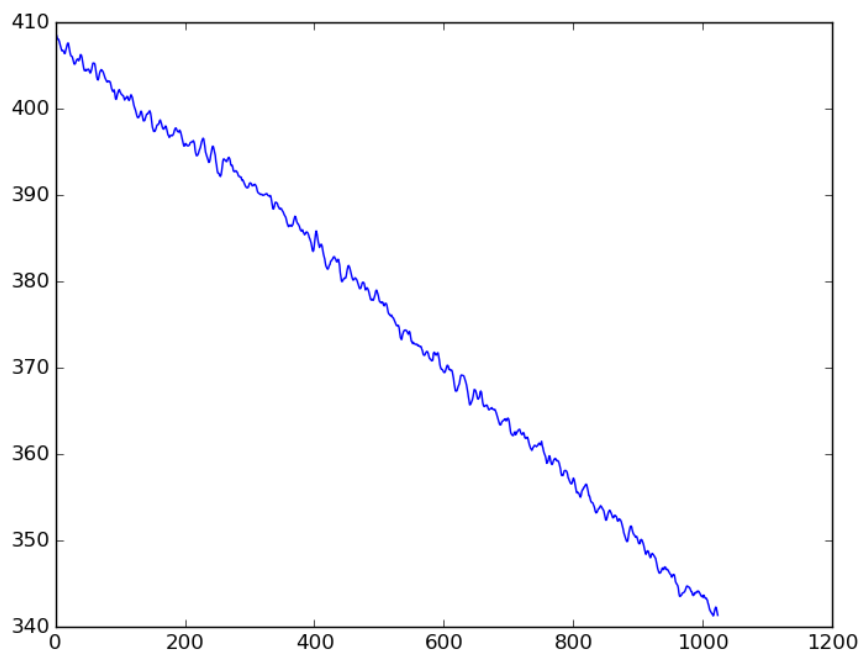


Figura 5.8: Ejemplo Samsung Tab 2 con Zoom

5.3. Evaluación de los Pesos de los Coeficientes de Interpolación

En este experimento se probó dispositivos de diferentes modelos, utilizando el algoritmo esperanza maximización, para detectar la matriz del kernel utilizado en los algoritmos de procesamiento de las imágenes en los diferentes dispositivos móviles.

Dispositivo	Recorte	Tasa de Acierto
EXP1_A	256	50%
EXP1_B	256	53.8%
EXP1_C	256	100%
EXP2_A	256	58.5%
EXP2_B	256	100%

Tabla 5.7 Resultados de la evaluación de los pesos de los coeficientes de interpolación sin SFFS.

En la Tabla 5.8 muestra los dispositivos móviles utilizados con 100 fotos por dispositivo, un recorte de 256x256 y en conjunto con el algoritmo SFFS.

Dispositivo	Recorte	Tasa de Acierto
EXP1_A	256	49.5%
EXP2_A	256	50.5%

Tabla 5.8 Resultados Experimento Esperanza Maximización 2 con SFFS.

En la Tabla 5.9 muestra los dispositivos móviles utilizados con 100 fotos por dispositivo, un recorte de 256x256 y un resultado de 76,4706%.

Marca	Modelo	Número de dispositivos
APPLE	Iphone 4 S	1
	Iphone 5	3
	Iphone 5 S	2
	Iphone 6	2
	Iphone 6 PLUS	1
	Iphone 7	2
HUAWEI	Y635-L01	2
	U8815	1
LG	E400	1
	NEXUS 5	4
SAMSUNG	A5_16	1
	Galaxy Nexus	1
	Galaxy S4 mini	1
	Galaxy S3	1
	Galaxy S5	1
	Galaxy S6	1
	Galaxy S Plus	1
	Galaxy S	1
	Galaxy Ace s5830	1
	Galaxy Ace s5830m	1
	Galaxy Grand 2	1
	SONNY	C2015
ST25A		1
ST25I		1
XPERIA_M2_D2303		1
Resultados		76,4706%

Tabla 5.9: Resultados Experimento Esperanza Maximización 3.

En la Tabla 5.10 se presentan los modelos APPLE con 100 fotos por dispositivo, un recorte de 256x256 y un resultado del 60%.

Marca	Modelo	Número de dispositivos
APPLE	Iphone 4 S	1
	Iphone 5	3
	Iphone 5 S	2
	Iphone 6	2
	Iphone 6 PLUS	1
	Iphone 7	2
Resultado		60%

Tabla 5.10: Resultados Experimento Esperanza Maximización Experimento Apple.

En la Tabla 5.11 con 50 fotos por dispositivo, un recorte de 256x256 y un resultado del 64,285%.

Marca	Modelo	Número de dispositivos
APPLE	Iphone 4	1
	Iphone 5	2
	Iphone 5 S	1
	Iphone 6	2
	Iphone 6 PLUS	2
	Iphone 7	2
	HUAWEI	Y635-L01
	U8815	1
LG	E400	1
	NEXUS 5	4
SAMSUNG	A5_16	1
	Galaxy Nexus	1
	Galaxy S4 mini	1
	Galaxy S3	1
	Galaxy S5	1
	Galaxy S6	1
	Galaxy S Plus	1
	Galaxy S	1
	Galaxy Ace s5830	1
	Galaxy Ace s5830m	1
	Galaxy Grand 2	1
Resultados		64,285%

Tabla 5.11: Resultados Experimento Esperanza Maximización EXPERIMENTO4_50.

En la tabla 5.12 se muestran los dispositivos móviles utilizados en el experimento. Por cada dispositivo se utilizaron 50 fotos por dispositivo, un recorte de 256x256 dando un resultado del 13,14%.

Marca	Modelo	Número de dispositivos
APPLE	Iphone 7	1
AQUARIS	M5	1
HUAWEI	Y635-L01	1
LG	Nexus 5	1
OUKITEL	K 6000 PRO	1
SAMSUNG	A5_16	1
XIAOMI	Redmi Note 2	1
Resultados		13,14%

Tabla 5.12: Resultados Experimento Esperanza Maximización EXPERIMENTO5.

Los resultados obtenidos en la identificación de la fuente se muestran en las Tablas 5.7 a la 5.12. Dado los resultados del experimento se puede concluir que puede obtenerse características relevantes en la identificación de la fuente de la imagen.

6. CONCLUSIONES Y TRABAJO FUTURO

6.1. Conclusiones

En este trabajo se ha propuesto un algoritmo de identificación de la fuente de imágenes de dispositivos móviles basado en la interpolación cromática. El algoritmo analiza los artefactos relacionados con el software del dispositivo que generó la imagen para extraer la huella presente en las imágenes consecuencia de los algoritmos de configuración CFA y las funciones utilizadas para la interpolación de color.

La evaluación de la propuesta se realizó con un dataset de 10000 fotos recolectadas directamente desde diferentes dispositivos, para garantizar las condiciones de captura de las fotografías.

Como se puede observar en los resultados de los experimentos, la tasa de identificación con la combinación del algoritmo de selección de característica es alta con un recorte 1024x1024. Sin embargo, en los casos en los que los experimentos incluyen solo móviles del mismo fabricante y modelo hay variaciones en los resultados lo que provoca que la identificación del dispositivo no sea la esperada. Estas variaciones se deben a la similitud del procesamiento de la imagen implementada en los mismos dispositivos. El algoritmo propuesto utiliza la máquina de soporte vectorial como sistema de clasificación.

Al realizar los experimentos con móviles del mismo fabricante y modelo y combinando las características extraídas con el algoritmo propuesto la tasa de acierto mejoró. Asimismo, los resultados mejoraron utilizando el algoritmo selección de características propuesto.

6.2. Trabajo Futuro

Como trabajos futuros pueden señalarse los siguientes:

- Extender el algoritmo de detección de interpolación a la autenticación de la fuente de la imagen: estimando el factor de interpolación en regiones de la imagen para detectar cambios en otras regiones.
- Crear de un gran dataset de imágenes que incluya el mayor número de fabricantes y los modelos posible para que sean accesible a otros investigadores.
- Desarrollar otros algoritmos de identificación de la fuente relacionados con los artefactos software del dispositivo para aumentar robustez del mismo.

RESUMEN EN INGLÉS

7. INTRODUCTION

7.1. Motivation

Currently, mobile devices are evolving very quickly to suit the needs of users and also the market, which results in an increase in the different models being sold and companies producing them. One of the needs that users have and look for on mobile devices is a built-in camera, which they always have at hand and avoids the need of carrying a separate camera [9]. This cases the companies to strive to improve the quality of the photo in the mobile devices. Thus, forensic computing is gaining a great deal of importance in the world of social networks because of the value of the information for business companies, government agencies and individuals [19]. By means of this data, we are able to obtain evidence in the form of images taken with these devices, which might have been be used for criminal purposes, and thereby identify the perpetrator.

7.2. Objectives

This research work, corresponding to the Trabajo de Fin de Grado (TFG), has the following objectives:

- Perform a study of the related work on the types of identification algorithms in the images in order to obtain a classification of the models.
- Carry out a study of the techniques of forensic analysis of digital image identification that exist in the literature in order to analyze and understand the most relevant techniques.
- Implement an algorithm to allow the determining whether a digital image comes from one of the selected devices.

7.3. Work Schedule

The project has been developed in 3 phases: Definition, Execution and Project Documentation. Activities in these phases are presented in Table 8.1.

During the first phase, we establish the objectives and scope of the Trabajo de Fin de Grado, the meetings with the team of tutors, and the follow-up of the work. Subsequently, during the Execution phase, the project was developed as defined in the previous stage. This phase consists of the following stages: specification of the requirements, design, implementation, and testing. During this phase, activities were carried out to monitor and control the progress of the project in order to supervise the activities carried out.

Finally, in the documentation phase, we conducted all documentation required for the realization of the Trabajo de Fin de Grado. In this phase was conducted together with the two previous phases.

8. CONCLUSIONS AND FUTURE RESEARCH

8.1. Conclusions

In this research work, we have proposed an algorithm for the identification of the source of images of mobile devices based on the chromatic interpolation. This algorithm analyzes artifacts related to the software of the device that generated the image. This way it is possible to extract the fingerprint in the images by means of the CFA configuration algorithms and the functions used for color interpolation.

The evaluation of the proposed strategy was carried out with a dataset of 10000 photos collected directly from different devices, to guarantee the conditions of capture of the photographs.

The results of the experiments showed that the identification rate with the combination of the algorithm for selection of the characteristic is high with a 1024x1024 cutout. However, in cases of the experiments that include only mobiles of the same manufacturer and model, there are variations in the results which causes that the identification of the device is other than expected. These variations are due to the similarity of image processing implemented in the same devices. The proposed algorithm uses the vector support machine as the classification system.

When running the experiments with mobiles of the same manufacturer and model, and combining the extracted characteristic with the proposed algorithm, the success rate improved. Likewise, the results were improved using the proposed algorithm for selection of the characteristics.

8.2. Future Research

Future research may include the following areas:

- Extend the interpolation detection algorithm to the authentication of image source: estimating the interpolation factor in regions of the image to detect changes in other regions.
- Create a large dataset of images that includes as many manufacturers and models as possible so that they are accessible to other researchers.
- Develop other algorithms for identification of the source related to device artifacts of the software in order to increase robustness of the identification process.

REFERENCIAS

- [AGRCISO⁺14] David Manuel Arenas González, Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Hiram Jafet Romo Torres, and Luis Javier Garcá Villalba. Identificación de la fuente en vdeos de dispositivos móviles. 2014.
- [AH09] Carlos Agualimpia and Rodrigo Hernández. Análisis Forense en Dispositivos Móviles con Symbian OS. *Documento de maestría, Dept. Ingeniería electrónica*, pages 1–6, 2009.
- [ALMB09] Jaime Alberto Echeverri Arias, Bell Manrique Losada, Francisco Javier Moreno, and Alejandro Bravo. Mejoramiento de imágenes usando funciones de base radial. *Revista Ingenierías Universidad de Medellín*, 8(15 Sup. 1):21–28, 2009.
- [AZ06] Marwan Al Zarouni. Mobile handset forensic evidence: a challenge for law enforcement. 2006.
- [Ber] Fernando Berzal. Clasificación y Predicción .
- [BSM06] Sevinc Bayram, Husrev Sencar, and Nasir Memon. Identifying digital cameras using cfa interpolation. In *IFIP International Conference on Digital Forensics*, pages 289–299. Springer, 2006.
- [BSM08] Sevinc Bayram, Husrev T Sencar, and Nasir Memon. Classification of Digital Camera-Models Based on Demosaicing Artifacts. *digital investigation*, 5(1):49–59, 2008.
- [BSMA05] Sevinc Bayram, H Sencar, Nasir Memon, and Ismail Avcibas. Source Camera Identification Based on CFA Interpolation. In *IEEE International Conference on Image Processing 2005*, volume 3, pages III–69. IEEE, 2005.
- [BSMA06] Sevinc Bayram, Husrev T Sencar, Nasir Memon, and I Avcibas. Improvements on Source Camera-Model Identification Based on CFA Interpolation. *Proc. of WG*, 11:24–27, 2006.
- [Cam13] Cambridge In Colour. Digital Image Interpolation, 2013.
- [CAS07] Oya Celiktutan, Ismail Avcibas, and Bülent Sankur. Blind Identification of Cellular Phone Cameras. In *Electronic Imaging 2007*, pages 65051H–65051H. International Society for Optics and Photonics, 2007.

- [CCL11] Chang-Hee Choi, Jung-Ho Choi, and Heung-Kyu Lee. CFA Pattern Identification of Digital Cameras Using Intermediate Value Counting. In *Proceedings of the Thirteenth ACM Multimedia Workshop on Multimedia and Security*, pages 21–26. ACM, 2011.
- [CESR12] Filipe de O Costa, Michael Eckmann, Walter J Scheirer, and Anderson Rocha. Open Set Source Camera Attribution. In *Graphics, Patterns and Images (SIBGRAPI), 2012 25th SIBGRAPI Conference on*, pages 71–78. IEEE, 2012.
- [CK08] Hong Cao and Alex C Kot. A Generalized Model for Detection of Demosaicing Characteristics. In *Multimedia and Expo, 2008 IEEE International Conference on*, pages 1513–1516. IEEE, 2008.
- [CK09a] Hong Cao and Alex C Kot. Accurate Detection of Demosaicing Regularity for Digital Image Forensics. *IEEE Transactions on Information Forensics and Security*, 4(4):899–910, 2009.
- [CK09b] Hong Cao and Alex C Kot. Accurate Detection of Demosaicing Regularity from Output Images. In *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pages 497–500. IEEE, 2009.
- [CK10] Hong Cao and Alex C Kot. Mobile Camera Identification Using Demosaicing Features. In *Circuits and systems (ISCAS), Proceedings of 2010 IEEE international symposium on*, pages 1683–1686. IEEE, 2010.
- [Eri07] Eric Dubois. Interpolation of RGB Components in Bayer CFA Images, 2007.
- [Gal05] Andrew C Gallagher. Detection of linear and cubic interpolation in jpeg compressed images. In *Computer and Robot Vision, 2005. Proceedings. The 2nd Canadian Conference on*, pages 65–72. IEEE, 2005.
- [GBK+01] Zeno J Geradts, Jurrien Bijhold, Martijn Kieft, Kenji Kurosawa, Kenro Kuroki, and Naoki Saitoh. Methods for Identification of Images Acquired with Digital Cameras. In *Enabling Technologies for Law Enforcement*, pages 505–512. International Society for Optics and Photonics, 2001.
- [GMR10] Yusleidy Alvarez Germade, Eduardo Barbará Morales, and Oscar Rodríguez Ramírez. Filtrado digital en el procesamiento de imágenes empleando matlab. *Convención Científica de Ingeniería y Arquitectura*, 2010.
- [HF10] Yizhen Huang and Na Fan. Learning from Interpolated Images Using Neural Networks for Digital Forensics. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 177–182. IEEE, 2010.

- [HPH14] Min Gu Hwang, Hyung Ju Park, and Dong Hwan Har. Source camera identification based on interpolation via lens distortion correction. *Australian Journal of Forensic Sciences*, 46(1):98–110, 2014.
- [HTRC16] Thanh Hai Thai, Florent Reiraint, and Rémi Coganne. Camera Model Identification Based on the Generalized Noise Model in Natural Images. *Digital Signal Processing*, 48:285–297, 2016.
- [int] Interpolation of RGB Components in Bayer CFA Images .
- [JF06] Micah K Johnson and Hany Farid. Exposing Digital Forgeries Through Chromatic Aberration. In *Proceedings of the 8th workshop on Multimedia and security*, pages 48–55. ACM, 2006.
- [JHF12] Qingyue Jin, Yizhen Huang, and Na Fan. Learning Images Using Compositional Pattern-Producing Neural Networks for Source Camera Identification and Digital Demographic Diagnosis. *Pattern recognition letters*, 33(4):381–396, 2012.
- [JWWABA16] Mehdi Jahanirad, Ainuddin Wahab Wahid Abdul, and Nor Badrul Anuar. An Evolution of Image Source Camera Attribution Approaches. *Forensic Science International*, 262:242–275, March 2016.
- [LFG06] Jan Lukas, Jessica Fridrich, and Miroslav Goljan. Digital Camera Identification from Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, 2006.
- [lib] Algoritmo de Selección de Características .
- [LLC+12] Qingzhong Liu, Xiaodong Li, Lei Chen, Hyuk Cho, Peter A. Cooper, Zhongxue Chen, Mengyu Qiao, and Andrew H. Sung. Identification of Smartphone-Image Source and Manipulation. In *Proceedings of the 25th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems*, pages 262–271, Dalian, China, June 2012.
- [ME11] Pablo Andrés Martínez Espinosa. Producto Multimedia de Turismo de Quito. 1:1–87, 2011.
- [MSGW08] Christine McKay, Ashwin Swaminathan, Hongmei Gou, and Min Wu. Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source. In *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, pages 1657–1660. IEEE, 2008.
- [Reb07] Jorge Valverde Rebaza. Detección de bordes mediante el algoritmo de canny. *Escuela Académico Profesional de Informática, Universidad Nacional de Trujillo (Perú)*, 2007.
- [SB13] Gastón Semprini and Alfredo Bozzetti. Informática Forense al Servicio de una Justicia Moderna. pages 210–221, Buenos Aires, Argentina, Agosto 2013.

- [SBSM07] Yagiz Sutcu, Sevinç Bayram, Husrev T Sencar, and Nasir Memon. Improvements on Sensor Noise Based Source Camera Identification. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 24–27. IEEE, 2007.
- [SCLW06] Kai San Choi, Edmund Y Lam, and Kenneth KY Wong. Feature Selection in Source Camera Identification. In *Systems, Man and Cybernetics, 2006. SMC'06. IEEE International Conference on*, volume 4, pages 3176–3180. IEEE, 2006.
- [SWL07] Ashwin Swaminathan, Min Wu, and KJ Ray Liu. Nonintrusive Component Forensics of Visual Sensors Using Output Images. *IEEE Transactions on Information Forensics and Security*, 2(1):91–106, 2007.
- [TWLY12] Min-Jen Tsai, Chen-Sheng Wang, Jung Liu, and Jin-Sheng Yin. Using Decision Fusion of Feature Selection in Digital Forensics for Camera Source Model Identification. *Computer Standards & Interfaces*, 34(3):292–304, 2012.
- [Uni14] Universidad de Oviedo. Interpolación de Imágenes, 2014.
- [VLCSK07] Tran Van Lanh, Kai-Sen Chong, Emmanuel Sabu, and Mohan S Kankanhalli. A Survey on Digital Camera Image Forensic Methods. In *2007 IEEE international conference on multimedia and expo*, pages 16–19. IEEE, 2007.
- [WAW+16] Jiaji Wu, Marco Anisetti, Wei Wu, Ernesto Damiani, and Gwanggil Jeon. Bayer Demosaicking With Polynomial Interpolation. *IEEE Transactions on Image Processing*, 25(11):5369–5382, 2016.
- [WKY09] Bo Wang, Xiangwei Kong, and Xingang You. Source Camera Identification Using Support Vector Machines. In *IFIP International Conference on Digital Forensics*, pages 107–118. Springer, 2009.
- [WTZ+15] Bo Wang, Yue Tan, Meijuan Zhao, Guo Yanqing, and Xiangwei Kong. Classifier Combination Based Source Identification for Cell Phone Images. *KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS*, 9(12):5087–5102, 2015.
- [WY06] Che Yen Wen and Kun Ta Yang. Image authentication for digital image evidence. *Forensic Science Journal*, 5(1):1–11, 2006.