UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA

DEPARTAMENTO DE INGENIERÍA DEL SOFTWARE E INTELIGENCIA ARTIFICIAL

TESIS DOCTORAL

Técnicas de identificación de la fuente de adquisición en imágenes digitales de dispositivos móviles

MEMORIA PARA OPTAR AL GRADO DE DOCTOR
PRESENTADA POR

David Manuel Arenas González

Director

Luis Javier García Villalba

Madrid, 2015

Técnicas de Identificación de la Fuente de Adquisición en Imágenes Digitales de Dispositivos Móviles

TESIS DOCTORAL

Memoria presentada para obtener el título de
Doctor por la Universidad Complutense de Madrid
en el Programa de Doctorado en Ingeniería Informática

David Manuel Arenas González

Director

Luis Javier García Villalba

Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid

Madrid, Marzo de 2015

Tesis Doctoral presentada por el doctorando David Manuel Arenas González en el Departamento de Ingeniería del Software e Inteligencia Artificial de la Universidad Complutense de Madrid para la obtención del título de Doctor por la Universidad Complutense de Madrid en el Programa de Doctorado en Ingeniería Informática.

Terminada en Madrid el 1 de Marzo de 2015.

Título:

Técnicas de Identificación de la Fuente de Adquisición en Imágenes Digitales de Dispositivos Móviles

Doctorando:

David Manuel Arenas González (darenas@fdi.ucm.es)
Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática
Universidad Complutense de Madrid
28040 Madrid, España

Director:

Luis Javier García Villalba (javiergv@fdi.ucm.es)

Esta tesis doctoral ha sido realizada dentro del grupo de investigación GASS (Grupo de Análisis, Seguridad y Sistemas, grupo 910623 del catálogo de grupos reconocidos por la UCM) como parte de las actividades de colaboración con la *School of Computing* y la *School of Physical Sciences* de la *University of Kent* (Canterbury, Kent, Reino Unido). La presente investigación ha sido financiada por el Ministerio de Defensa (a través del proyecto artículo 83 LOU UCM 321/2011), por la Agencia Española de Cooperación Internacional para el Desarrollo del Ministerio de Asuntos Exteriores y de Cooperación (a través del proyecto A1/037528/11) y por la empresa Safelayer Secure Communications S. A. (a través del proyecto artículo 83 LOU UCM 307/2013). Parte de los cálculos de este trabajo fueron realizados en EOLO, el sistema de computación de alto rendimiento del Clúster de Cambio Global y Nuevas Energías del Campus de Excelencia Internacional (CEI) Campus Moncloa, financiado por el Ministerio de Educación, Cultura y Deporte y por el Ministerio de Ciencia e Innovación. Esto es una contribución al CEI Moncloa.

Agradecimientos

Sin ninguna duda, estaré siempre agradecido a Javier por haberme dado la posibilidad de realizar esta tesis. Sin su ayuda, conocimientos y capacidad de organización esta tesis no habría sido posible. También le agradezco todos los medios que ha puesto a mi disposición para facilitar la comunicación en la distancia y el desarrollo de los distintos experimentos de este trabajo.

No puedo olvidarme de agradecer especialmente el apoyo recibido de Ana Lucila. Ella ha sido uno de los pilares más importantes en mi formación como investigador. Sin ella, igualmente, esta tesis no habría sido posible.

Agradezco la colaboración de todos los miembros del Grupo de Análisis, Seguridad y Sistemas (GASS). Ellos han sido espejo de situaciones análogas a la vividas por mí en el proceso de elaboración de este trabajo. También han sido fuente de ánimo en los momentos más difíciles.

Asimismo, agradezco al Vicerrectorado de Innovación de la Universidad Complutense de Madrid las facilidades de procesamiento ofrecidas.

No puede faltar el agradecimiento a mis hermanos Álvaro y Rafael, así como a mis padres Antonio y Antonia, por su ánimo, paciencia y comprensión. Gracias a ellos he podido seguir adelante y finalizar esta tesis.

Esta tesis doctoral ha sido realizada dentro del grupo de investigación GASS (Grupo de Análisis, Seguridad y Sistemas, grupo 910623 del catálogo de grupos reconocidos por la UCM) como parte de las actividades de colaboración con la *School of Computing* y la *School of Physical Sciences* de la *University of Kent* (Canterbury, Kent, Reino Unido).

Parte de los cálculos de este trabajo fueron realizados en EOLO, el sistema de computación de alto rendimiento del Clúster de Cambio Global y Nuevas Energías del Campus de Excelencia Internacional (CEI) Campus Moncloa, financiado por el Ministerio de Educación, Cultura y Deporte y por el Ministerio de Ciencia e Innovación. Esto es una contribución al CEI Moncloa.

Abstract

The number of integrated digital cameras on mobile devices and their use in everyday life is continuously growing. Daily large number of images generated by these devices are circulating on the Internet or are used as evidence or proof in judicial proceedings. As a consequence, forensic analysis of digital images of mobile devices becomes important in many real-life situations. It is noteworthy that forensics specific images techniques are required for mobile devices, not to be valid in most cases, the techniques used for the *Digital Still Cameras* (DSCs), because there are significant intrinsic features which differentiate both types of cameras. Also, the quality of the elements that conform them is different, being usually better in the DSCs. This work deals with the design of several techniques of image source acquisition identification for digital images generated with mobile devices.

In the first place *Theia* is presented, a tool that allows the processing of *Exchangeable Image File Format* (Exif) metadata for digital image source acquisition identification. *Theia* allows advanced management of independent projects with different image sets and it has different functionality at the level of image processing, both individually and in groups. The image processing in groups allows the forensic analyst to perform different analyzes of large image databases quickly, easily and with great versatility. *Theia* has been compared with other similar tools, being the only one which includes image processing in groups and tamper detection based on processing the thumbnail.

Secondly, after a Exif metadata manual binary analysis performed with *Theia*, it is shown that the manufacturers do not often follow the specification, finding nine different types of anomalies. This is crucial, because it creates interoperability problems in applications that manipulate metadata. *Theia* is the only application of forensic analysis that detects these type of anomalies in following the Exif specification.

In the third place two techniques of digital image source acquisition identification are developed. In both techniques it has been successfully used *Support Vector Machine* (SVM) classifiers. These techniques are used in what are known as *closed scenarios*. In this kind of scenarios the images, whose source acquisition must be determined, belong to a group of devices known beforehand. Therefore, the image source acquisition identification is enclose to a certain known number of devices. The first technique works with a set of 25 features based on sensor noise and wavelet transform. The second technique uses a set of 81 features based on the sensor photo response non-uniformity.

These identification techniques have different settings that allow to adapt the use of several features sets for various purposes. It is also possible to apply them to the source type identification (computer, mobile device or DSC).

The two identification techniques presented are closely related, since both are based on the use of image content features. *Theia* allows to use different feature sets with a great versatility. Therefore, the forensic analyst for the same image set, can perform different analyzes using different feature sets combinations. Depending on issues such as the number or type of devices, the characteristics of them, if there are devices of the same brand, etc., the forensic analyst will use the most appropriate settings.

Keywords: Exif, Exif anomalies, forensic analysis, image classification, image source acquisition identification, metadata, mobile device, Photo Response Non-Uniformity, PRNU, sensor noise, sensor pattern noise, support vector machine, SVM, Theia, wavelet transform.

Resumen

El número de cámaras digitales integradas en dispositivos móviles así como su uso en la vida cotidiana está en continuo crecimiento. Diariamente gran cantidad de imágenes generadas por este tipo de dispositivos circulan en Internet o son utilizadas como evidencias o pruebas en procesos judiciales. Como consecuencia, el análisis forense de imágenes digitales de dispositivos móviles cobra importancia en multitud de situaciones de la vida real. Cabe destacar que se necesitan técnicas forenses específicas para imágenes de dispositivos móviles, no siendo válidas en la mayoría de los casos las técnicas utilizadas para las DSCs, debido a que existen notables características intrínsecas que diferencian ambas tipos de cámaras. Asimismo, la calidad de los elementos que las conforman es distinta, siendo usualmente mejor en las DSCs. Este trabajo versa sobre el diseño de diversas técnicas para la identificación de la fuente de adquisición de imágenes digitales generadas con dispositivos móviles.

En primer lugar se presenta *Theia*, una herramienta que permite el tratamiento de los metadatos Exif para la identificación de la fuente de adquisición de imágenes digitales. *Theia* permite la gestión avanzada de proyectos independientes con distintos conjuntos de imágenes y tiene distintas funcionalidades a nivel de tratamiento de imágenes, tanto individualmente como en grupo. El tratamiento grupal de imágenes permite al analista forense realizar de forma rápida, sencilla y con gran versatilidad diversos análisis sobre grandes bancos de imágenes. *Theia* ha sido comparada con otras herramientas similares, siendo la única que incluye el tratamiento grupal y la detección de manipulaciones basadas en el procesado de la imagen en miniatura.

En segundo lugar se muestra, tras un análisis binario manual de los metadatos Exif realizado con *Theia*, que los fabricantes no siguen frecuentemente la especificación, encontrándose nueve diferentes tipos de anomalías. Este hecho es fundamental, creando problemas de interoperabilidad en las aplicaciones que manipulan los metadatos. *Theia* es la única aplicación de análisis forense que detecta este tipo de anomalías en el seguimiento de la especificación Exif.

En tercer lugar se desarrollan dos técnicas de identificación de la fuente de adquisición en imágenes digitales. En ambas técnicas se han utilizado de forma exitosa clasificadores SVM. Estas técnicas se aplican en lo que se denominan *escenarios cerrados*. En este tipo de escenarios las imágenes, cuya fuente de adquisición hay que determinar, pertenecen a un grupo de dispositivos conocidos a priori. Por tanto, la identificación de la fuente de adquisición de las imágenes está acotada a un número de dispositivos determinado y conocido. La primera técnica trabaja con un conjunto de 25 características basadas en el ruido del sensor y en la transformada wavelet. La segunda técnica emplea un conjunto de 81 características basadas en el patrón de no-uniformidad de la foto-respuesta del sensor.

Estas técnicas de identificación presentan distintas configuraciones que permiten adaptar el uso de los diferentes conjuntos de características a distintas finalidades. Es posible además aplicarlas a la identificación del tipo de dispositivo fuente (computador, dispositivo móvil o DSC).

Las dos técnicas de identificación presentadas tienen estrecha relación, ya que ambas están basadas en el uso de características del contenido de la imagen. *Theia* permite utilizar los diversos conjuntos de características con total versatilidad. Por tanto, el analista forense, para un mismo conjunto de imágenes, puede realizar distintos análisis utilizando diferentes combinaciones de los conjuntos de características. Dependiendo de aspectos como el número o tipo de dispositivos, sus características, si existen dispositivos de la misma marca, etc., el analista forense utiliza la configuración más adecuada.

Palabras clave: Análisis forense, anomalías Exif, clasificación de imágenes, dispositivo móvil, Exif, identificación de fuente de adquisición, máquinas de soporte vectorial, metadatos, patrón de no-uniformidad de la foto-respuesta del sensor, patrón del ruido del sensor, PRNU, ruido del sensor, SVM, Theia, transformada wavelet.

Índice General

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Lista de Acrónimos

1NN	*First Nearest Neighbor*.
A-GPS	*Assisted Global Positioning System*.
APP0	*Application Marker Segment 0*.
APP1	*Application Marker Segment 1*.
APPn	*Application Marker Segment n*.
ASCII	*American Standard Code for Information Interchange*.
BMP	*Bitmap*.
CCD	*Charge Coupled Device*.
CFA	*Color Filter Array*.
CIPA	*Camera and Imaging Products Association*.
CMOS	*Complementary Metal Oxide Semiconductor*.
CYGM	*Cyan-Yellow-Green-Magenta*.
CYYM	*Cyan-Yellow-Yellow-Magenta*.
DCT	*Discrete Cosine Transform*.
DHT	*Define Huffman Table*.
DIP	*Digital Image Proccesor*.

DQT *Define Quantization Table.*

DSC *Digital Still Camera.*

EC *Error Cumulants.*

EM *Expectation-Maximization.*

EOI *End of Image.*

ERE *Eigenfeature Regularization.*

Exif *Exchangeable Image File Format.*

FPN *Fixed Pattern Noise.*

GIF *Graphics Interchange Format.*

GPS *Global Positioning System.*

GRGB *Green-Red-Green-Blue.*

HTML *HyperText Markup Language.*

ICC *International Color Consortium.*

IFD *Image File Directory.*

IIM *Information Interchange Model.*

IPTC *International Press Telecommunications Council.*

IQM *Image Quality Metrics.*

ITU *International Telecommunication Union.*

JEITA *Japan Electronics and Information Technology Industries Association.*

JFIF *JPEG File Interchange Format.*

JPEG *Joint Photographic Experts Group.*

MAP *Maximum A-Posteriori Probability.*

MD5 *Message-Digest Algorithm 5.*

MMS *Multimedia Messaging System.*

MOS *Metal Oxide Semiconductor.*

MP3 *MPEG-1 Audio Layer III.*

NGS *Normalized Group Sizes.*

PDF *Portable Document Format.*

PIL *Python Imaging Library.*

PNG *Portable Network Graphics.*

PNU *Pixel Non-Uniformity.*

PRNU *Photo Response Non-Uniformity.*

PSD *Photoshop Document.*

PSVM *Probabilistic Support Vector Machine.*

QMF *Quadrature Mirror Filter.*

RAM *Random-Access Memory.*

RBF *Radial Basis Function.*

RDF *Resource Description Framework.*

RGB *Red-Green-Blue.*

RGBE *Red-Green-Blue-Emerald.*

RGBW *Red-Green-Blue-White.*

RMS *Root Mean Square.*

ROI *Region of Interest.*

SFFS *Sequential Floating Forward Selection.*

SMS *Short Message Service.*

SOF *Start of Frame.*

SOI *Start of Image.*

SOS *Start of Stream.*

SPN *Sensor Pattern Noise.*

SVM *Support Vector Machine.*

TIFF *Tagged Image File Format.*

W3C *World Wide Web Consortium.*

WAV *Waveform Audio File Format.*

XML *Extensible Markup Language.*

XMP *Extensible Metadata Platform.*

Parte I

Descripción de la Investigación

Capítulo 1

Introducción

Actualmente, la demanda de dispositivos móviles (teléfonos móviles, *smartphones*, tabletas, etc.) aumenta año tras año. La industria de este tipo de dispositivos ha desarrollado la tecnología necesaria para abaratar los costes y hacerlos, de esta forma, más accesibles al público. Concretamente para el caso de los *smartphones*, en 2013 las ventas crecieron un 42,3 % con respecto al año anterior, superando por primera vez en número de ventas a los teléfonos móviles tradicionales [Inc14]. Para el caso de las *tablets*, en 2013 las ventas aumentaron un 68 % con respecto al año anterior [Gar14].

Según estimaciones de la *International Telecommunication Union* (ITU), en 2014 existirán 6,87 miles de millones de líneas dadas de alta en dispositivos móviles en todo el mundo, lo que supone un incremento sobre los 6,62 miles de millones de suscripciones de 2013 y los 6,19 miles de millones de 2012. Asimismo, la ITU estima que en 2014 habrá 2,29 miles de millones de altas de líneas de banda ancha en dispositivos móviles, suponiendo un notable aumento sobre los 1,91 y 1,54 miles de millones de altas de los años 2013 y 2012, respectivamente.

En la Figura 1.1 se muestra la evolución del incremento de las altas de dispositivos móviles en distintas zonas del mundo a lo largo de los últimos 10 años.

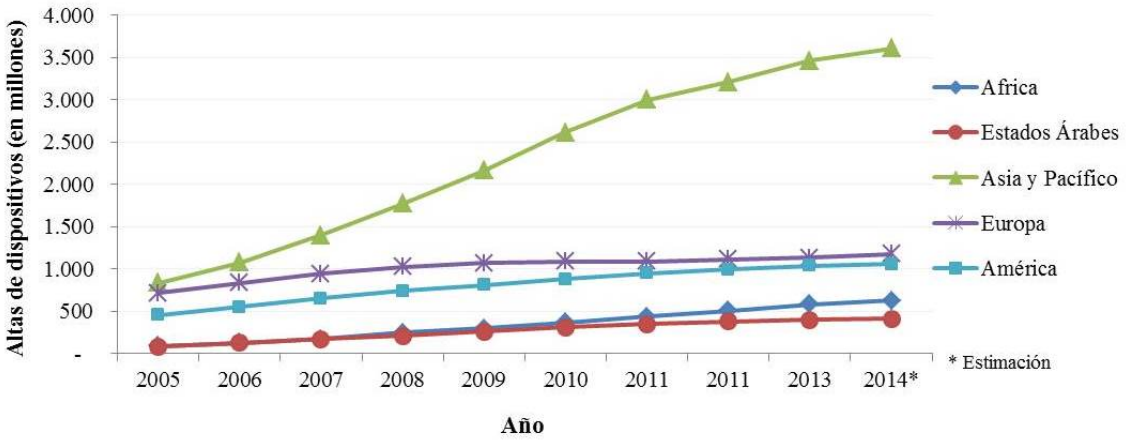


Figura 1.1: Evolución del incremento de suscripciones de dispositivos móviles

Una vez descrita esta visión general en cifras de la magnitud de la presencia de dispositivos móviles en el mundo, no debe pasarse por alto la irrupción de este tipo de dispositivos en el día a día. Entre otros factores, el incremento de las capacidades de almacenamiento, procesamiento y usabilidad de los dispositivos móviles han permitido que estén presentes en diversas actividades, lugares y eventos de la vida cotidiana. Tanto es así, que un gran número de personas tienen y usan más de un dispositivo móvil y un usuario típico en promedio consulta su móvil unas 150 veces al día, siendo 8 de ellas para hacer uso de la funcionalidad de la cámara [AM12].

En los países industrializados el 97 % de teléfonos móviles incorpora una cámara digital integrada. Asimismo, la mayor parte del resto de tipos de dispositivos móviles también posee una cámara digital integrada. Estas cámaras, a diferencia de las DSCs, son llevadas por sus dueños gran parte del tiempo a la mayoría de los lugares a los que asisten [AM14]. En el año 2016 la venta de DSCs descenderá de un 47 % de cuota de mercado sobre el total de cámaras digitales que obtuvo en el 2012 a un 27 %. Asimismo, se prevé un incremento en las ventas de cámaras digitales integradas en teléfonos móviles, PC y tabletas, de un 31 % de cuota de mercado sobre el total de cámaras digitales en 2012, a un 42 % en el 2016 [IC 13].

En la Figura 1.2 se muestra una previsión para 2016 de la cuota de mercado de las cámaras digitales para distintas utilidades o sectores [IC 13].

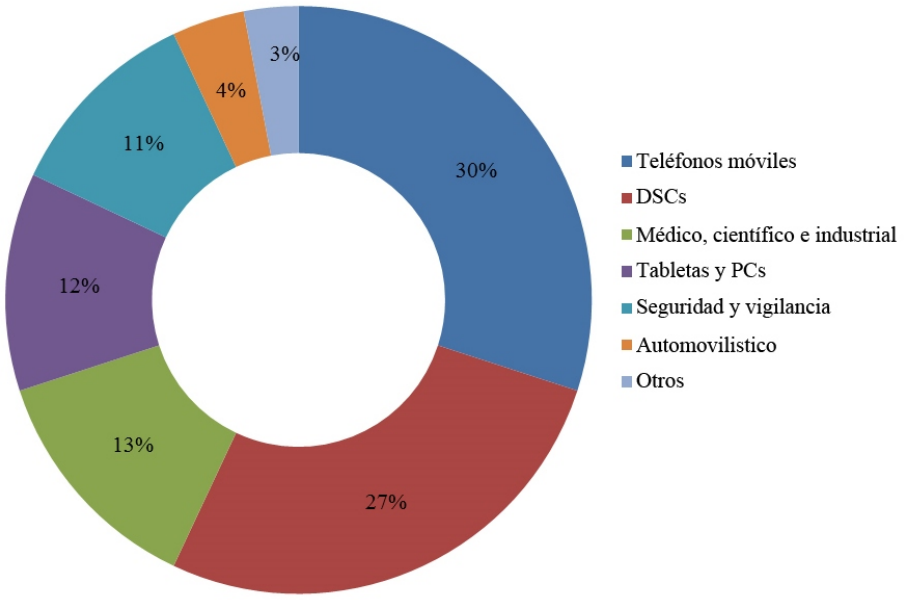


Figura 1.2: Previsión de cuota de mercado de cámaras digitales para 2016

Del mismo modo existen predicciones que indican que las DSCs desaparecerán en pro de las nuevas cámaras digitales integradas en dispositivos móviles [Bae10], ya que el aumento de calidad de éstas crece a un ritmo imparable.

Prácticamente la totalidad de estas cámaras digitales tienen funciones de grabación de vídeo. Actualmente, existe una gran competencia entre los fabricantes por integrar una videocámara de alta definición al alcance del usuario en todo momento. Como consecuencia, y dada la gran cantidad de tiempo que una persona pasa junto a los dispositivos móviles, éstos se han convertido para muchas personas en el primer dispositivo de captura de fotografías y grabación de vídeos.

El amplio uso de cámaras digitales en dispositivos móviles es una realidad en la vida cotidiana. Diariamente pueden verse imágenes generadas por dispositivos móviles en telenoticias, distintas aplicaciones, correo electrónico o en redes sociales. Webs como *Facebook*, *Youtube* o *Twitter* entre otras, se sitúan en los puestos más altos de la lista de webs más visitadas, siendo una parte considerable de su contenido capturado con cámaras digitales de dispositivos móviles [ALE14].

Todo esto hace que en ciertos casos existan restricciones legales o limitaciones a su utilización en distintos lugares, tales como: colegios, universidades, oficinas de gobierno, empresas, etc. Además, y como consecuencia de todo lo anterior, cada día las imágenes digitales generadas con dispositivos móviles son más utilizadas como testigos silenciosos en procesos judiciales (pornografía infantil, espionaje industrial, violencia callejera, redes sociales, ...), siendo en muchos casos piezas cruciales de la evidencia de un crimen [AZ06]

[WY06]. Por todas estas razones el análisis forense de imágenes digitales de dispositivos móviles cobra especial fuerza en la actualidad. El estudio debe ser concreto para este tipo de dispositivos, ya que poseen características específicas que permiten obtener mejores resultados, no siendo válidas las técnicas forenses para imágenes digitales generadas por otros tipos de dispositivos. En [TNC10] se describe de forma clara y razonada la necesidad de técnicas de análisis forense específicas para dispositivos móviles.

1.1 Identificación del Problema

Las imágenes digitales generadas con dispositivos móviles no son más que archivos digitales almacenados en un soporte codificados en un determinado formato. Además, en la mayoría de los casos estos archivos contienen información adicional al propio contenido visual de la escena denominada "metadatos".

Una vez generados estos archivos, pueden ser movidos de un soporte a otro con excesiva facilidad, ya sea o no de forma malintencionada. Por tanto, puede darse el caso de perder fácilmente la pista del dispositivo que generó la imagen o, dicho de otra forma, perder la referencia de su fuente de adquisición.

Los archivos de imágenes digitales pueden ser modificados además de una forma más o menos elaborada por cualquier usuario. Existen una gran cantidad de programas de edición accesibles a cualquier tipo de usuario que permiten modificar este tipo de contenido digital siendo muchas veces estos cambios imperceptibles para el ojo humano. Igualmente al caso anterior, estos cambios pueden ser intencionados o malintencionados, pero independientemente de la fe con la que se realizó el cambio, la imagen pierde su originalidad con respecto a la generación por parte de la fuente de adquisición.

Estas situaciones pueden generar problemas o indefiniciones cuando las imágenes son utilizadas como evidencias en algún proceso, ya sea judicial o no, dado que no se puede garantizar la identificación de la fuente de adquisición del contenido o la no manipulación del mismo sin realizar un análisis forense previo.

Descritos los problemas, en este trabajo se proponen distintas soluciones de análisis forense centrándose principal y mayormente en el problema de la identificación de la fuente de adquisición de una imagen. Dentro de este aspecto se proponen soluciones basadas en los metadatos y en el contenido de la propia imagen. Las soluciones basadas en los metadatos aportan información adicional al analista forense. Asimismo, se presenta una aplicación, *Theia*, que, en base a los metadatos, permite gestionar grandes bancos de imágenes con distintas funcionalidades adicionales. Las distintas soluciones presentadas se han creado teniendo en cuenta, entre otros aspectos, un enfoque de aplicación práctica y la realización de experimentos con una amplia variedad de situaciones y dispositivos implicados.

1.2 Resumen de las Contribuciones

Los resultados de la investigación realizada en esta Tesis comprenden diversas contribuciones que han sido publicadas en diferentes revistas/congresos de alto impacto. Como se representa en la Figura 1.3, estas contribuciones se enmarcan en el área del análisis forense de imágenes digitales de dispositivos móviles. *Theia* constituye el Capítulo 6 y es una herramienta para el análisis forense de imágenes de dispositivos móviles que recopila las técnicas de identificación de la fuente de adquisición desarrolladas en esta tesis. Un análisis de los metadatos Exif clasificado por categorías [SOAGGVHC12a] y un análisis de las anomalías Exif clasificadas por tipos de errores encontrados [SOAGGVHC12b] [SOAGGVHC14] se realizan en el Capítulo 7. Los algoritmos de identificación de la fuente de adquisición de imágenes de dispositivos móviles se encuentran organizados en dos capítulos. El algoritmo basado en las características del ruido del sensor y la transformada wavelet se presenta en el Capítulo 8 [SOAGRC+14]. Por último, el algoritmo basado en el patrón de no-uniformidad de la foto-respuesta del sensor se describe en el Capítulo 9 [RCAGSO+13] [RCAGSOGV14] [SOGVAG+15].

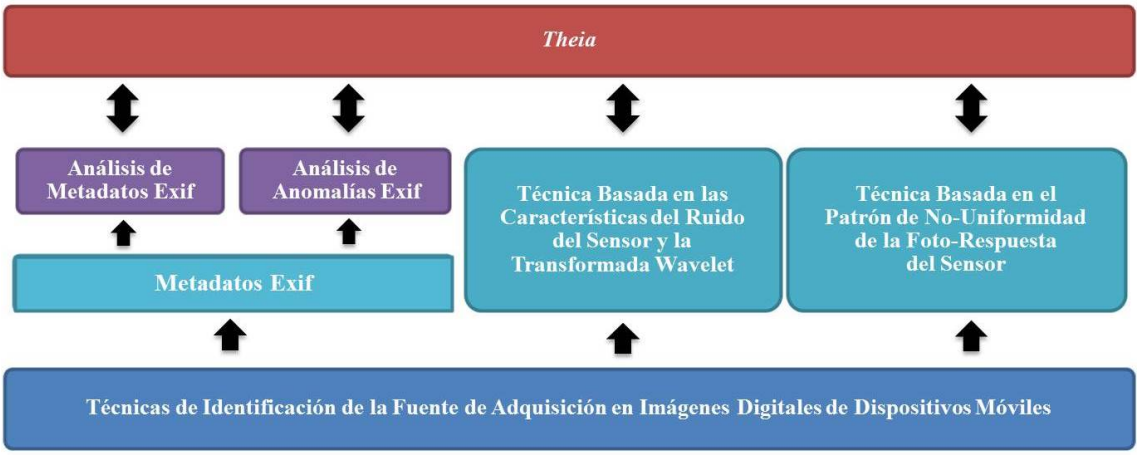


Figura 1.3: Contribuciones de la Tesis

1.3 Estructura del Trabajo

Esta Tesis se estructura como sigue:

El Capítulo 2 describe el proceso de adquisición y creación de imágenes en diferentes tipos de dispositivos. Inicialmente, se presenta el proceso de adquisición de imágenes en cámaras digitales, haciendo hincapié sobre la matriz *Color Filter Array* (CFA) y los distintos tipos de sensores. El objetivo es señalar los elementos principales del proceso de

adquisición de imágenes que difieren entre una DSC y una cámara digital de un dispositivo móvil.

El Capítulo 3 presenta una visión general de los metadatos en imágenes digitales. Se realiza una breve descripción de las especificaciones de metadatos *Tagged Image File Format* (TIFF), *JPEG File Interchange Format* (JFIF), *International Press Telecommunications Council* (IPTC) y *Extensible Metadata Platform* (XMP). Asimismo, se muestran los resultados del análisis binario de los metadatos Exif de imágenes reales, mostrándose ejemplos de las estructuras anteriormente descritas. Este análisis binario manual es de gran importancia, ya que mejora la comprensión en profundidad de los conceptos generales sobre metadatos que se tratan en los siguientes capítulos.

El Capítulo 4 introduce las razones que justifican la necesidad del análisis forense en dispositivos móviles. Asimismo, se indican las principales características que hacen a los dispositivos móviles como fuentes potenciales de análisis forense y se exponen las distintas ramas del análisis forense en las imágenes digitales. Finalmente, se hace una descripción panorámica de los distintos tipos de técnicas forenses para la identificación de la fuente de adquisición de imágenes digitales.

El Capítulo 5 resume las principales técnicas de análisis forense para la identificación de la fuente de adquisición de imágenes digitales y los principales trabajos relacionados. Se comentan los trabajos más relevantes sobre los distintos tipos de técnicas: basadas en metadatos, basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en las características de las imágenes y basadas en el uso de las imperfecciones del sensor. También contiene un cuadro comparativo resumen los trabajos relacionados tratados en este capítulo, destacando los temas de especial relevancia.

El Capítulo 6 especifica una herramienta para el análisis forense de imágenes digitales denominada *Theia*. En primer lugar se muestra la estructura del diseño y aspectos relativos a la implementación de la aplicación. A continuación, se describen las principales características de la herramienta y sus funcionalidades con respecto al tratamiento de metadatos Exif. Finalmente, se realiza una comparación con otras herramientas de análisis de metadatos Exif.

El Capítulo 7 presenta distintos análisis de metadatos Exif realizados a bancos de imágenes con *Theia*. En primer lugar, se hace un análisis de los metadatos Exif teniendo en cuenta las distintas categorías de metadatos [SOAGGVHC12a]. En segundo lugar se efectúa un análisis de las anomalías en los metadatos Exif, clasificando las mismas en los 9 tipos de errores más comúnmente encontrados [SOAGGVHC14] [SOAGGVHC12b]. Finalmente, se dan las conclusiones generales de los análisis efectuados a los distintos bancos de imágenes, los cuales destacan por tener un gran número de imágenes de una amplia variedad de modelos de dispositivos móviles.

El Capítulo 8 desarrolla un algoritmo basado en las características del ruido y la transformada wavelet para la identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles [SOAGRC+14]. Inicialmente se presentan los conceptos generales para la comprensión del algoritmo. Asimismo, se señalan las características y configuración de las máquinas SVM utilizadas para la clasificación, las cuales serán utilizadas en todas las demás técnicas de identificación de la fuente de adquisición. El capítulo finaliza con la presentación de los distintos experimentos realizados sobre bancos de imágenes de dispositivos móviles.

El Capítulo 9 describe un algoritmo basado en las características del patrón de no-uniformidad de la foto-respuesta del sensor para la identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles [RCAGSO+13] [RCAGSOGV14] [SOGVAG+15]. Cabe destacar que este algoritmo posee distintos parámetros de configuración (independientes del método de clasificación o preprocesamiento de las imágenes), los cuales han sido utilizados en los distintos experimentos con el objetivo de evaluar las tasas de acierto en función de los mismos.

El Capítulo 10 expone las principales conclusiones de este trabajo, así como algunas posibles líneas futuras de investigación.

Capítulo 2

Imágenes Digitales

El objetivo general de este capítulo es facilitar la comprensión del proceso de adquisición y creación de imágenes en diferentes tipos de dispositivos. En primer lugar se presenta el proceso de adquisición de imágenes en cámaras digitales, haciendo especial referencia a la matriz CFA y a los distintos tipos de sensores. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

2.1 Introducción

El proceso de adquisición y creación de imágenes digitales, también conocido como *pipeline*, generalmente posee notables diferencias entre los distintos tipos de dispositivos. Dentro del mismo tipo de dispositivo la estructura del *pipeline* es semejante y varía en pequeños aspectos por factores como el fabricante, la calidad de la cámara o las prestaciones que ofrece.

A continuación se muestra la estructura general del proceso de generación de una imagen en cámaras digitales. Dentro del apartado de cámaras digitales se hace hincapié en los aspectos relevantes relacionados con las cámaras digitales de dispositivos móviles.

2.2 Proceso de Adquisición en Cámaras Digitales

Aunque muchos de los detalles del proceso de generación de una imágen en una cámara digital pertenecen a cada fabricante y a cada tipo de dispositivo (y se mantienen como información confidencial), existe una estructura general del mismo para todas ellas. A grandes rasgos una cámara fotográfica digital se compone de un sistema de lentes, un grupo de filtros, una matriz de filtros de colores o CFA, un sensor de imagen y un procesador

de imagen o *Digital Image Proccesor* (DIP) [BSM08]. Gráficamente, la estructura general del *pipeline* de una cámara digital puede resumirse en la Figura 2.1

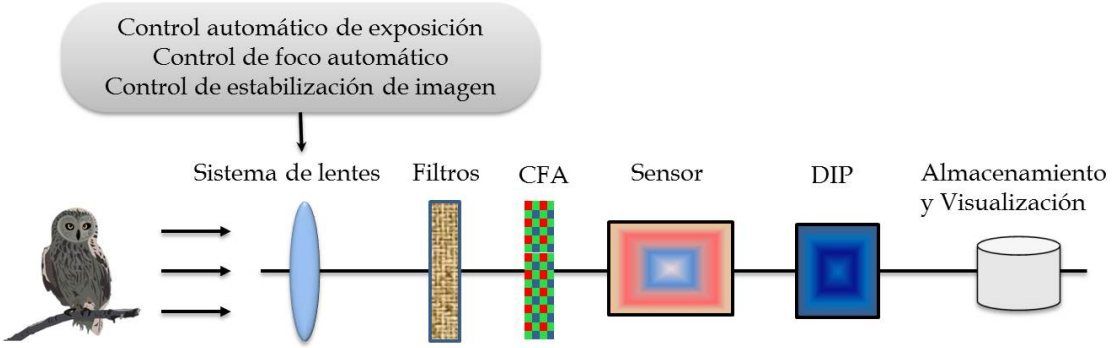


Figura 2.1: Proceso de adquisición de imágenes en cámaras digitales

Primeramente un sistema de lentes capta la luz de la escena controlando la exposición, el foco y la estabilización de la imagen. Después, la luz llega a un grupo de filtros (*anti-aliasing* e infrarrojo, entre otros) que mejoran la calidad de la imagen.

Seguidamente la luz pasa al sensor de imagen. Puede haber mecanismos que interactúen con el sensor para determinar la exposición (tamaño de apertura, velocidad de obturación, control de ganancia automático) y la distancia focal de la lente. El sensor de imagen es una matriz de elementos sensibles a la luz llamados píxeles, los cuales son monocromáticos. Cada elemento de esta matriz de píxeles toma la luz incidente y genera una señal analógica proporcional a la intensidad de la luz recibida. Esta señal analógica se convierte a una señal digital y se transmite al procesador de imagen.

A la hora de capturar una imagen es necesario medir tres o más bandas para cada píxel. Dada la monocromaticidad de los sensores se necesitarían idealmente tres sensores para cada píxel. Esto elevaría notablemente el coste de la cámara. Para solucionar esto en la práctica las cámaras normalmente usan un único sensor de imagen para cada píxel junto a una matriz CFA que se coloca antes del sensor para producir los distintos colores básicos.

Una vez que el procesador de imagen recibe la señal digital generada por el sensor, se elimina el ruido y otras anomalías introducidas en las señales digitales (*artifacts*), con la finalidad de obtener una imagen visualmente más agradable. El más destacado de estos procesos es el denominado interpolación cromática (*demosaicing* o *demosaicking*). El *demosaicing* es el proceso más complejo desde el punto de vista computacional y las técnicas utilizadas suelen ser propiedad del fabricante de la cámara. Este algoritmo utiliza los valores de los píxeles vecinos para obtener todos los canales que no han sido medidos (recuérdese que el sensor en cada píxel sólo detecta el canal que deja pasar la matriz CFA).

Posteriormente, se realizan procesos como la corrección de píxeles defectuosos, el balanceo de blancos y la corrección gamma. La corrección de píxeles defectuosos originados por imperfecciones en el sensor corrige estos píxeles mediante interpolación. El balanceo de blancos permite una reproducción más fiel del color, sin que haya colores dominantes que son especialmente notables en tonos neutros como el blanco. La corrección *gamma* ajusta los valores de intensidad de la imagen. Aunque los algoritmos para llevar a cabo estos procesos están presentes en todas las cámaras, los detalles exactos de la forma de realizarlos pueden variar entre los diferentes fabricantes, e incluso, entre los distintos modelos de un mismo fabricante.

Finalmente, la imagen generada por el procesador se comprime. En las cámaras de dispositivos móviles normalmente se utiliza el algoritmo *Joint Photographic Experts Group* (JPEG) [Ham] para ahorrar espacio, almacenándose en la memoria permanente del dispositivo junto con los metadatos de la imagen (en prácticamente la totalidad de los casos en formato Exif [RSYD05]).

2.2.1 Matriz de Filtros de Color

La matriz CFA es una de las partes más importantes en el *pipeline* de una cámara [APS98]. La matriz CFA se encuentra sobre el sensor monocromo y su función es adquirir la información del color de la escena. Cada celda del filtro de color deja pasar la luz de acuerdo a un rango de longitudes de onda, de tal manera que las intensidades filtradas separadas incluyen información sobre el color de la luz, como se ilustra en un ejemplo en la Figura 2.2.

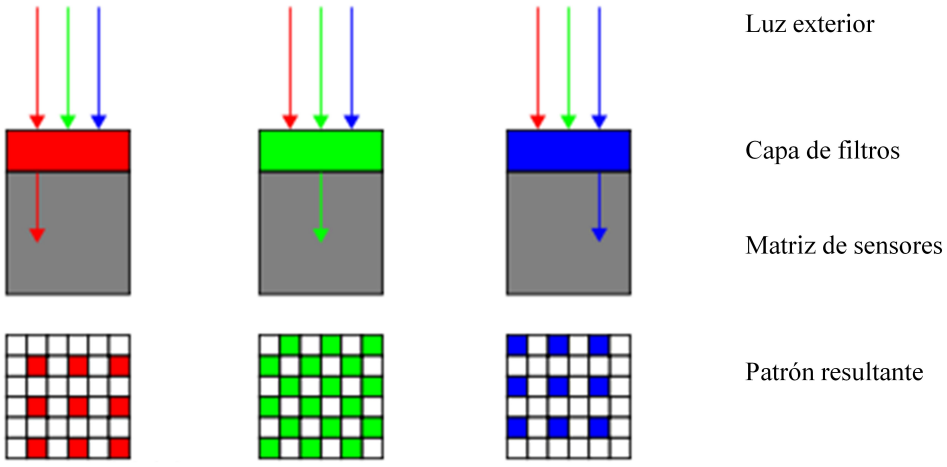
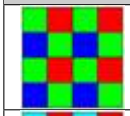
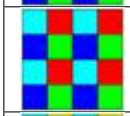
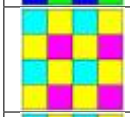
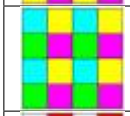
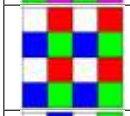
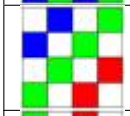
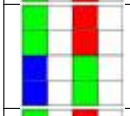
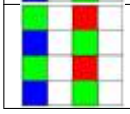


Figura 2.2: Matriz CFA

El algoritmo de *demosaicing* es el encargado de obtener los valores que faltan para cada uno de los colores del filtro CFA mediante interpolación. Es decir, utiliza los valores de los píxeles vecinos para obtener todos los valores que no han sido medidos. El diseño del filtro de color puede variar de un fabricante a otro. El algoritmo de interpolación puede variar también para cada fabricante, aún utilizando el mismo tipo de matriz CFA. Asimismo, la elección de la matriz CFA puede influir en la nitidez y apariencia final de la imagen. Existen distintos patrones CFA como el modelo *Green-Red-Green-Blue* (GRGB) de Bayer mostrado en la Figura 2.2. Otros modelos de patrón CFA son el filtro *Red-Green-Blue-Emerald* (RGBE), *Cyan-Yellow-Yellow-Magenta* (CYYM), *Cyan-Yellow-Green-Magenta* (CYGM) o el *Red-Green-Blue-White* (RGBW). En la Tabla 2.1 se muestran los filtros de color citados anteriormente [Nak05].

Tabla 2.1: Tipos de filtros de color

Filtro	**Nombre**	**Descripción**	**Tamaño del patrón (en píxeles)**
	Filtro Bayer	Es el más común: 1 píxel azul, 1 rojo y 2 verdes.	2x2
	Filtro RGBE	Como el de Bayer pero con uno de los píxeles verdes de color esmeralda. Usado en algunas cámaras Sony.	2x2
	Filtro CYYM	1 píxel cian, 2 amarillos y 1 magenta. Usado en algunas cámaras Kodak.	2x2
	Filtro CYGM	1 píxel cian, 1 amarillo, 1 verde y 1 magenta. Usado en pocas cámaras.	2x2
	Filtro Bayer RGBW	Como el de Bayer pero con uno de los píxeles verdes de color blanco.	2x2
	RGBW #1	Tres ejemplos de filtros RGBW de Kodak con el 50 % de los píxeles blancos.	4x4
	RGBW #2		4x4
	RGBW #3		4x4

En la Figura 2.3 puede verse un ejemplo de captura de color con el filtro GRGB del patrón de Bayer y generación de la imagen final por parte del algoritmo de interpolación o *demosaicing*. Este filtro captura para el canal rojo el 25 % de los píxeles, para el verde el 50 % y para el azul el 25 % restante. Esto significa que para la construcción de la imagen final se tiene que recuperar el 75 % de los píxeles del canal rojo, el 50 % del canal verde y el 75 % del canal azul.

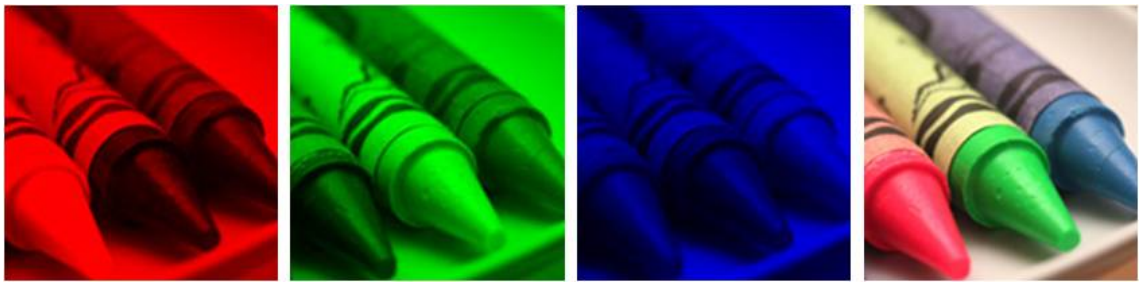


Figura 2.3: Ejemplo de aplicación del filtro GRGB de Bayer a una imagen real

2.2.2 Tipos de Sensores

Como se comentó anteriormente, el sensor es el componente que se encarga de captar la luz y generar una señal digital en función de su intensidad. Actualmente existen dos tipos de tecnologías utilizadas para la fabricación de sensores de cámaras digitales: *Charge Coupled Device* (CCD) y *Complementary Metal Oxide Semiconductor* (CMOS). Ambos tipos de sensores están formados esencialmente por semiconductores de metal-óxido (*Metal Oxide Semiconductor* (MOS)), distribuidos en forma de matriz y funcionando de forma similar. Sin embargo, hay características que diferencian a estas tecnologías.

La diferencia clave entre las dos tecnologías de sensores es el lugar en el que se digitalizan los píxeles y la forma en la que se lleva a cabo la lectura de las cargas. Los sensores CCD poseen una estructura muy sencilla pero necesitan contar con un chip adicional para tratar la información de salida del sensor, produciendo sistemas más grandes y costosos. En contraste, los sensores CMOS poseen un diseño de píxeles activos independientes. Se les denomina activos porque ellos mismos realizan la digitalización, ofreciendo mayor velocidad y reduciendo el coste y tamaño de los sistemas. Todos los píxeles de una matriz CCD captan la luz simultáneamente, lo cual propicia una salida más uniforme, a diferencia de los sensores CMOS que realizan la lectura generalmente como barrido progresivo (evitando el efecto *blooming*). Los sensores CCD son muy superiores a los CMOS en el rango dinámico y en términos de ruido; en contrapartida los sensores CMOS son más sensibles a la luz y, en condiciones de poca iluminación, se comportan mejor.

En sus inicios los sensores CMOS no eran considerados tan buenos como los sensores CCD. Sin embargo, la tecnología CCD ha llegado a su límite y actualmente la tecnología CMOS está desarrollándose y superando continuamente sus deficiencias. Tanto es así que existen los denominado sensores *Smart* CMOS [Oht08] con el objetivo de mejorar las deficiencias de los sensores CCD y CMOS convencionales. La mayoría de las DSCs utilizan sensores CCD. En los dispositivos móviles, por el contrario, es más común el uso de sensores CMOS.

2.3 Síntesis del Capítulo

El objetivo de este capítulo ha sido introducir a grandes rasgos el proceso de adquisición de imágenes digitales. Se ha comenzado con la exposición del *pipeline* de imágenes en cámaras digitales. Se ha realizado especial hincapié en los elementos del *pipeline* que pueden ser objeto de estudio para futuras técnicas forenses de identificación de la fuente de adquisición. Finalmente, se ha realizado una descripción de los distintos tipos de sensores, con el objetivo de aclarar las diferencias que hay en este elemento entre DSCs y cámaras digitales de dispositivos móviles.

Capítulo 3

Metadatos en Imágenes Digitales

Este capítulo muestra una visión general de los metadatos en imágenes digitales. Comienza con una introducción general al concepto de los metadatos en imágenes digitales. Seguidamente se ha descrito detalladamente la especificación de metadatos Exif. Posteriormente, se presentan brevemente las especificaciones de metadatos TIFF, JFIF, IPTC y XMP. Finalmente, se realiza un análisis binario de los metadatos Exif de imágenes reales, mostrándose ejemplos de las estructuras anteriormente descritas. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

3.1 Generalidades

Los metadatos habitualmente son denominados "datos sobre los datos", es decir, información de interés que complementa el contenido principal de un documento digital. Las imágenes digitales son almacenadas en una gran variedad de formatos como TIFF, JPEG y *Photoshop Document* (PSD). Cada formato de imagen tiene distintas reglas de cómo los metadatos son almacenados junto al propio archivo que contiene la imagen. Algunos de los distintos contenedores de metadatos para los distintos formatos son: *Image File Directory* (IFD) Exif/TIFF, Adobe XMP e IPTC-IIM. Cada uno de estos contenedores de metadatos tiene un formato propio que indica las propiedades de los metadatos que son almacenados, el orden y su codificación en el contenedor. En cada contenedor suele haber una subdivisión con criterios semánticos. Estos grupos semánticos se dividen a su vez en propiedades de metadatos individuales. Cada propiedad tiene asociada unos tipos de datos específicos como pueden ser cadenas de caracteres, números o vectores. Algunas propiedades como la orientación de la imagen no son comunes a los distintos contenedores estándar; en cambio, otras, como las cadenas de *copyright*, pueden ser almacenadas por varios contenedores con similar información pero posiblemente con una semántica o estructura sutilmente distinta (Figura 3.1).

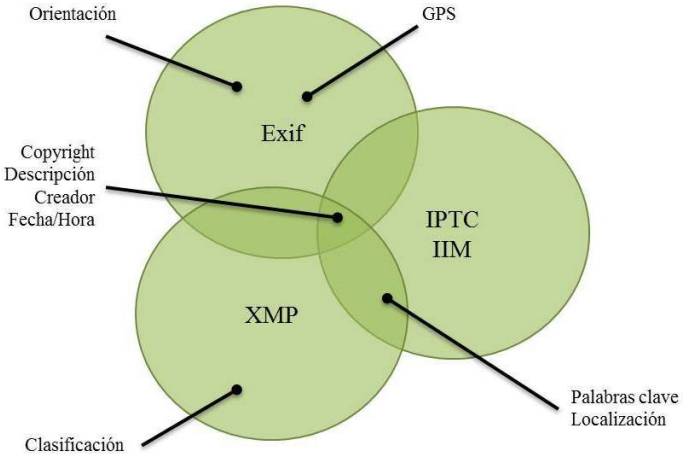


Figura 3.1: Contenedores de metadatos

Esta complejidad estructural hace que tradicionalmente cause problemas en el uso eficiente de los metadatos. Algunos de los más destacados son:

- Los diversos dispositivos y aplicaciones tratan las especificaciones de metadatos ambiguamente o con definiciones incompletas de diferentes formas. Asimismo, toman distintas políticas a la hora de almacenar la información de los metadatos.
- Los dispositivos y aplicaciones pueden almacenar metadatos propietarios, denominados *Maker Notes*, dentro de los contenedores. Esta práctica es débil y problemática ya que estos datos pueden perderse fácilmente cuando otra aplicación modifique el archivo.
- Algunas aplicaciones utilizan las propiedades de forma inadecuada para fines específicos diferentes para los que fueron creadas. Esto crea problemas de compatibilidad entre las distintas aplicaciones que utilizan correctamente las propiedades siguiendo las especificaciones establecidas.

Estos problemas causan en los usuarios frustración y desconfianza sobre los diferentes sistemas de metadatos, ya que lo que se busca es la interoperabilidad entre los distintos productos y servicios de imagen digital. Conscientes de estos problemas los fabricantes invierten gran cantidad de recursos para resolverlos. Tanto es así que existen grupos de fabricantes como el *Metadata Working Group* con el objetivo de mitigar o erradicar los problemas anteriormente descritos. Este grupo está formado por empresas como *Apple*, *Adobe*, *Canon*, *Microsoft*, *Nokia* y *Sony*. Incluso con la existencia de este grupo, los problemas no se resuelven por completo, dada la inmensa variedad de fabricantes existentes. Cabe destacar que este hecho no implica la inutilidad del uso de metadatos en imágenes, ya que actualmente se puede asegurar que son imprescindibles e inseparables en una imagen digital.

3.2 Formato Exif

El estándar Exif ha sido conjuntamente desarrollado por dos asociaciones: *Japan Electronics and Information Technology Industries Association* (JEITA) y *Camera and Imaging Products Association* (CIPA). El formato Exif define un conjunto de etiquetas TIFF para describir imágenes fotográficas. La especificación usa los formatos de archivos existentes como JPEG [Ham] y TIFF Rev. 6.0 [Ado92] a los que se agregan etiquetas específicas de metadatos. No está soportado en JPEG 2000 o *Portable Network Graphics* (PNG). Los metadatos Exif se pueden clasificar en cuatro grandes categorías [RCC+08]:

- Información relacionada con la fecha y hora de diferentes eventos.
- Características técnicas de configuración de la cámara. Ésta incluye información estática como el modelo de cámara y el fabricante, e información que varía con cada imagen como la orientación, apertura, velocidad del obturador, distancia focal y medidor de exposición.
- Información sobre localización, que puede provenir de un *Global Positioning System* (GPS) conectado a la cámara.
- Descripción e información sobre el *copyright*.

3.2.1 Estructura General del Formato JPEG

Existen distintas versiones de la especificación de Exif. Cada dispositivo soporta una versión que incluye a todas las anteriores. La versión Exif utilizada es una etiqueta más incluida en los propios metadatos. La última versión de la especificación es la 2.3 de abril de 2010 [Com13].

Dado que el formato más utilizado en cámaras digitales, y concretamente en dispositivos móviles, es JPEG, a continuación se describen los elementos y estructuras de datos que utiliza JPEG/Exif.

Todos los archivos JPEG comienzan con el valor binario '0xFFD8' (*Start of Image* (SOI)) y terminan con '0xFFD9' (*End of Image* (EOI)). SOI y EOI son marcadores que no tienen datos posteriores a diferencia de los otros que tienen una estructura fija y datos asociados. En un marcador con datos asociados el campo "tamaño de los datos" sigue la alineación de bytes denominada "Motorola" (*big-endian*). Es importante destacar que en el tamaño de los datos se incluyen los dos bytes que indican el propio tamaño.

En el formato JPEG la marca '0xFFDA' (*Start of Stream* (SOS)) indica el inicio de los datos propiamente dichos de la imagen, cuyo fin se limita con la marca EOI. Un esquema

general con la posibilidad de n marcadores para una imagen JPEG se presenta en la Tabla 3.1.

Tabla 3.1: Esquema general con marcadores de una imagen JPEG

SOI	**Marcador (1 a n)**				**SOS**			**Datos Imagen**	**EOI**
FFD8	FF	No. de Marca (1 byte)	Tamaño de los datos (2 bytes)	Datos (n bytes)	FFDA	Tamaño de los datos (2 bytes)	Datos (n bytes)	Datos (n bytes)	FFD9

3.2.2 Estructura de Datos Exif

Una vez vista la estructura general a grandes rasgos de un archivo JPEG se va a pasar a un grado más de concreción para mostrar donde se encuentran ubicados los metadatos Exif. La estructura general de un archivo JPEG/Exif se muestra en la Tabla 3.2.

Tabla 3.2: Estructura general de un archivo JPEG/Exif

SOI
APP1 (no excede de 64Kb)
APP2 (debe ser almacenado en esta posición si es necesario y puede haber varios)
APPn (no son utilizados por Exif; n es un valor entre 3 y 15, ambos incluidos)
DQT
DHT
DRI
SOF
SOS
Datos de la imagen
EOI

Los marcadores obligatorios en un archivo JPEG/Exif son: SOI, *Application Marker Segment 1* (APP1), *Define Quantization Table* (DQT), *Define Huffman Table* (DHT), *Start of Frame* (SOF), SOS y EOI. Además, es obligatorio que estén los datos comprimidos de la imagen propiamente dicha.

La información Exif es albergada en el segmento APP1. Existe un conjunto de segmentos *Application Marker Segment n* (APPn) no utilizados por Exif, que pueden ser empleados por los fabricantes para almacenar cualquier otro tipo información manteniendo la compatibilidad con Exif.

Exif utiliza el marcador APP1 para evitar conflictos con el marcador *Application Marker Segment 0* (APP0) del formato JFIF. Tras el tamaño del segmento APP1, se encuen-

tra la cadena "Exif" en caracteres *American Standard Code for Information Interchange* (ASCII) ('0x45786966') seguida de 2 bytes ('0x00'), que indica que ese archivo sigue la especificación Exif.

Tras el marcador APP1, pueden existir otros marcadores JPEG. La estructura básica del segmento APP1 se presenta en la Tabla 3.3.

Tabla 3.3: Estructura general del segmento APP1 de una imagen JPEG/Exif

APP1 Marker
APP1 Length
Exif Identifier Code
TIFF Header
0th IFD
0th IFD Value
1st IFD
1st IFD Value
1st IFD Image Data

Exif utiliza la estructura TIFF para almacenar los datos, que posee 2 IFDs, el 0th IFD y el 1st IFD. El 0th IFD contiene información sobre la propia imagen y el 1st IFD se utiliza para almacenar todo lo relacionado con el *thumbnail* (imagen en miniatura).

Por tanto, tras el indicador "Exif" (con sus dos bytes a '0x00' posteriores) vienen los datos de cabecera TIFF (primeros 8 bytes del formato), que tienen la siguiente estructura:

- **Definición del tipo de alineación de los datos**: Lo definen los dos primeros bytes. Este dato es muy importante y siempre tiene que tenerse en cuenta. Existen dos opciones:
  - *'0x4949' ("II")*: Alineación "Intel", es decir, alineación *little-endian*. Por ejemplo, el valor en decimal 232167 ('0x038AE7') se almacena como '0x03-0x8A-0xE7'.
  - *'0x4D4D' ("MM")*: Alineación "Motorola", es decir, *big-endian*. Por ejemplo, el valor en decimal 232167 ('0x038AE7') se almacena como '0xE7-0x8A-0x03'.

  Aunque JPEG siempre utiliza la alineación "Motorola", Exif permite los dos tipos de alineaciones.

- **Bytes fijos**: Los siguientes dos bytes siempre tienen un valor fijo '0x2A00'. Es importante recordar que hay que tener en cuenta el tipo de alineación. Si es "MM" se almacena como '0x2A00' y si es "II" como '0x002A'.

- **Desplazamiento**: Los últimos 4 bytes de la cabecera TIFF indican el desplazamiento (*offset*) al primer IFD cuya estructura se define posteriormente. Este desplazamiento se cuenta a partir del primer byte del tipo de alineación. Habitualmente el primer IFD comienza inmediatamente después de la cabecera TIFF, por lo que el valor suele ser '0x00000008'. Un esquema general de la cabecera TIFF puede observarse en la Tabla 3.4.

Tabla 3.4: Esquema general de la cabecera TIFF

Alineación (2 bytes)	**Marca fija (2 bytes)**	**Desplazamiento al primer IFD (4 bytes)**
"II" ó "MM"	0x2A00	0xLLLLLLLL

3.2.2.1 Directorio del Fichero de la Imagen

Un IFD está compuesto por los siguientes campos: 2 bytes que indican el número de entradas del directorio, entradas del directorio con un tamaño de 12 bytes y un desplazamiento de 4 bytes al siguiente IFD. La estructura básica de un IFD se muestra en la Tabla 3.5.

Tabla 3.5: Estructura básica de un IFD

No. de Entradas (2 bytes)	**Entradas del directorio (0xYYYYXXXXNNNNNNNNDDDDDDDD) (12 bytes * 0xTTTT)**				**Desplazamiento al siguiente IFD** (4 bytes)
0xTTTT	Etiqueta (2 bytes)	Tipo de datos (2 bytes)	No. de elementos (4 bytes)	Valor del desplazamiento (4 bytes)	0xLLLLLLLL

Cada entrada del directorio (12 bytes) tiene la siguiente estructura:

- **Etiqueta**: Son los dos primeros bytes. Los identificadores de las etiquetas en Exif 0th IFD y 1st IFD son los mismos que los de la especificación TIFF. El orden de las etiquetas en un directorio no está especificado en Exif.
- **Tipos de datos**: Siguientes dos bytes. En Exif 2.3 [Com10] los tipos de datos son:
  - BYTE - 1: Entero sin signo de 8 bits (1 byte).
  - ASCII - 2: Un byte (8 bits) que contiene caracteres ASCII de 7 bits. Esta cadena es terminada en NULL ('0x00').
  - SHORT - 3: Entero sin signo de 16 bits (2 bytes).
  - LONG - 4: Entero sin signo de 32 bits (4 bytes).
  - RATIONAL - 5: Dos LONGs. El primero es el numerador y el segundo es el denominador. Por tanto, este tipo ocupa 64 bits (8 bytes).

  - UNDEFINED - 7: Tipo byte que puede tomar cualquier valor dependiendo de la definición o significado del campo.
  - SLONG - 8: Un entero con signo de 32 bits (4 bytes) en notación complemento a 2.
  - SRATIONAL - 9: Dos SLONGs. El primero es el numerador y el segundo es el denominador. Por tanto, este tipo de dato ocupa 64 bits (8 bytes).

- **Número de elementos**: Siguientes 4 bytes. Indica el número de elementos que almacena la etiqueta. Es muy importante destacar que el número de elementos es algo totalmente diferente al número de bytes. Por ejemplo, si este campo es '0x00000004' y el tipo es LONG, en los datos se almacenan 4 LONGs, con lo cual la longitud es 16 bytes y no 4.

- **Valor del desplazamiento**: Siguientes 4 bytes. Si el valor es '0x00000000' quiere decir que es el último IFD. En este campo hay que tener en cuenta dos casos:

  - Si el tamaño de los datos a almacenar es menor o igual a 4 bytes, este campo almacena directamente los datos.
  - Si el tamaño de los datos a almacenar es mayor de 4 bytes, este campo almacena el desplazamiento donde se encuentran los datos con respecto al comienzo de la cabecera TIFF.

3.2.3 Información de Imagen en Miniatura

El formato Exif permite que el archivo contenga una imagen en miniatura (*thumbnail*), utilizada para la indexación de la imagen principal. La propia especificación Exif 2.3 no obliga a que todas las imágenes tengan *thumbnail*, pero sí lo recomienda. Esta imagen puede estar en algún formato comprimido o descomprimido, independientemente de que el formato de la imagen principal sea comprimido (JPEG). El *thumbnail* se incluye en el 1st IFD de distinta forma, dependiendo de si se almacena en un formato comprimido o descomprimido. Para evitar duplicar definiciones, el 1st IFD no se utiliza para almacenar etiquetas que posean información TIFF de la imagen o información guardada en otra parte como si fuera cualquier otro marcador JPEG.

3.3 Formato TIFF

TIFF es un formato de archivo basado en etiquetas para el almacenamiento e intercambio de imágenes [Ado92]. La primera versión de la especificación TIFF fue publicada por la Corporación Aldus en otoño de 1986, tras una serie de encuentros con una serie de

desarrolladores software y fabricantes de escáneres, aunque su versión más reciente es TIFF 6.0 publicada en 1992 por Adobe Systems, que es el actual responsable de la especificación. El propósito de TIFF es describir y almacenar datos de la imagen para proporcionar un entorno con el que las aplicaciones puedan intercambiar datos de la misma.

Los metadatos son un componente esencial del formato TIFF. Los metadatos TIFF se componen básicamente de tres grupos de etiquetas: *Baseline*, *Extension* y *Private*. El conjunto de metadatos TIFF es extensible para propósitos particulares mediante etiquetas privadas. Uno de los conjuntos de etiquetas privadas más destacados son los Exif, formato de metadatos que se ha descrito anteriormente.

Aunque el sistema de metadatos TIFF ha tenido mucho éxito, la proliferación de conjuntos de etiquetas privadas nuevos complica la extracción de los metadatos. Muchos de los programas para extraer metadatos TIFF sólo obtienen las etiquetas pertenecientes a los grupos *Baseline* y *Extension*. Dentro de las etiquetas privadas únicamente el conjunto Exif es ampliamente soportado y utilizado por los distintos fabricantes y aplicaciones. Otras de las características más relevantes de TIFF son:

- TIFF incluye varios esquemas de compresión, que permiten a los desarrolladores elegir el más apropiado para sus aplicaciones.
- No está unido a dispositivos electrónicos específicos.
- Es portable, no favoreciendo a un sistema operativo particular ni a un sistema de archivos, compilador o procesador concreto.
- Está diseñado para ser ampliable y evolucionar según las nuevas necesidades lo requieran.

3.4 Formato JFIF

JFIF es un formato de metadatos que contiene las imágenes guardadas en compresión JPEG [Ham]. Permite el intercambio de metadatos entre una gran variedad de plataformas y aplicaciones. Es un formato simple, que no incluye algunas de las características avanzadas de otros formatos de archivo de intercambio de imágenes. Formalmente los estándares Exif y JFIF son incompatibles, ya que ambos especifican que sus segmentos de aplicación deben de ir los primeros en el archivo de imagen. En la práctica muchas aplicaciones producen archivos con ambos segmentos, pero esto puede crear problemas.

3.5 Formato IPTC

IPTC es un consorcio de grandes agencias de noticias y publicidad. Ha desarrollado y mantiene un estándar técnico para mejorar el intercambio de noticias. En 1979 el primer estándar del IPTC era solo texto y fue definido para proteger los intereses de la industria de las telecomunicaciones. Después, en 1991, se creó un nuevo estándar, el *Information Interchange Model* (IIM). IIM es un formato para transmitir documentos de noticias en texto, fotografías y otros tipos de archivos multimedia. El IIM define las llamadas cabeceras IPTC, que actualmente existen en gran cantidad de archivos de imágenes. Estas cabeceras tienen una compleja estructura de datos donde se almacena un conjunto de definiciones de metadatos. Estas cabeceras son insertadas por software como, por ejemplo, Adobe Photoshop.

La información IPTC se encuentra separada en distintos registros cada uno de los cuales tienen sus propias etiquetas. Los registros IPTC son: IPTC EnvelopeRecord, IPTC ApplicationRecord, IPTC NewsPhoto, IPTC PreObjectData, IPTC ObjectData e IPTC PostObjectData. Adobe creó XMP en 2001 y los estándares IPTC adoptaron XMP como sucesor del IIM en 2005. Éste es ampliamente utilizado por los profesionales de la imagen digital. Información como el nombre del fotógrafo, el título de la imagen, la información de *copyright*, etc., pueden incluirse de forma manual o automática.

3.6 Formato XMP

XMP [XMP13] es una tecnología de etiquetado basada en *Extensible Markup Language* (XML) que permite incluir metadatos en el propio archivo. Con XMP las aplicaciones de escritorio y sistemas de publicidad poseen un método para capturar y compartir información aprovechando los metadatos insertados. XMP estandariza la definición, la creación y el procesamiento de los metadatos.

XMP define un modelo de metadatos que puede ser utilizado con cualquier otro conjunto de metadatos definido. XMP también define un particular esquema de propiedades básicas muy útiles para el almacenamiento de la historia de un recurso así como de su procesamiento.

Para el almacenamiento de los datos utiliza un subconjunto del *World Wide Web Consortium* (W3C) *Resource Description Framework* (RDF). Sin embargo, los usuarios pueden definir sus propias propiedades, es decir, XMP permite a cada programa o dispositivo a lo largo de la vida del archivo añadir su propia información.

Los metadatos XMP son más flexibles que los demás y se adaptan a más usuarios. Se apoyan en los Dublin Core, que se componen de un conjunto de 15 elementos originalmente

concebidos para la descripción generada por el autor de recursos en la web, si bien también se emplean en bibliotecas y museos. Se crearon los IPTC Core para facilitar la transición de IPTC/IIM hacia XMP. El IPTC Core es una transferencia explícita de los valores de los metadatos de las cabeceras IPTC al marco de trabajo XMP. Pocos programas pueden visualizar los metadatos XMP.

XMP puede ser utilizado en diversos formatos de archivos como *Portable Document Format* (PDF), JPEG, JPEG 2000, *Graphics Interchange Format* (GIF), PNG, *Hyper-Text Markup Language* (HTML), TIFF, Adobe Illustrator, PSD, *MPEG-1 Audio Layer III* (MP3), Audio Video Interleave, *Waveform Audio File Format* (WAV), RF64, Audio Interchange File Format, PostScript, Encapsulated PostScript. En un archivo JPEG la información XMP suele ser incluida junto con los metadatos Exif e IPTC.

3.7 Análisis Binario de Imágenes de Dispositivos Móviles

Una vez presentada la especificación Exif y teniendo en cuenta que es la utilizada en la mayoría de los dispositivos móviles y DSCs [RCC+08], se ha estimado oportuno realizar un análisis a nivel binario de varias imágenes tomadas con dispositivos móviles. Este análisis tiene como objetivos profundizar en el conocimiento de la propia especificación y comprobar si ésta es seguida por los fabricantes.

Dado el alto número de etiquetas que posee Exif y que cada imagen sólo posee un subconjunto de ellas, se han elegido algunas estructuras y etiquetas para el análisis. Éste ha seguido un orden lógico de estructuras de mayor a menor nivel: estructura general JPEG, cabecera TIFF, marcadores, IFD y etiquetas concretas.

Para el primer análisis se han seleccionado dos fotografías tomadas desde 2 teléfonos móviles (Samsung Galaxy S y Sony Ericsson W580i). Estas fotografías no han sufrido ningún tipo de proceso posterior a la captura de la imagen en el teléfono móvil.

Inicialmente se comprobó que los archivos son JPEG. Analizando a grandes rasgos su estructura general se observa que ambas imágenes comienzan con el valor binario "0xFFD8" (SOI) y terminan con el valor binario "0xFFD9" (EOI).

Posteriormente, en la imagen del Samsung Galaxy S puede comprobarse la existencia del marcador APP1 ("0xFFE1"), seguido de su tamaño "0x288E" (alineación "Motorola"), es decir, 10882 bytes de datos (incluidos los 2 bytes que indican la longitud). Por tanto, APP1 en este caso comienza en "0x0004" y termina en "0x2892" (este byte no incluido).

Para el caso del Sony Ericsson W580i, igualmente se contempla el marcador APP1 ("0xFFE1"), seguido del tamaño "0x133D" (alineación "Motorola"), es decir, 4925 bytes de datos (incluidos los 2 bytes que indican la longitud). Por tanto, APP1 en este caso

comienza en "0x0004" y termina en "0x1341" (este byte no incluido).

Si se extrae para las dos imágenes el siguiente marcador de APP1 se observan diferentes resultados:

- **Samsung Galaxy S**: El siguiente marcador (en la dirección "0x2892") es "0xFFDB", que según Exif se corresponde con DQT.
- **Sony Ericsson W580i**: El siguiente marcador (en la dirección "0x1314") es "0xFFC4", que según Exif se corresponde con DHT.

Con estos dos datos anteriores se observa que tras APP1, en imágenes diferentes siguen marcadores distintos, lo cual está permitido por Exif. Dentro de la estructura del marcador APP1 se encuentran los datos de la cabecera TIFF, donde puede observarse que ambas imágenes siguen la especificación Exif, tienen alineación "Intel" y desplazamiento "0x00000008" al primer IFD.

Una vez analizados algunos marcadores, hay que pasar al siguiente nivel: los IFDs. En la imagen del Samsung Galaxy S se va a examinar la estructura de su primer IFD y las dos primeros etiquetas. Tras la cabecera TIFF se encuentran los bytes "0x0C00". Teniendo en cuenta que la alineación es "Intel", estos bytes indican cuantas entradas tiene el directorio actual, en este caso el 0th IFD. Por tanto, el 0th IFD tiene 12 entradas. La primera entrada del directorio "0x0E010200140000009E000000" se interpreta como se muestra en la Tabla 3.6.

Tabla 3.6: Análisis de etiquetas del 0th IFD.

Dispositivo Móvil	Primera entrada 0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento
Samsung Galaxy S	0x0E010200140000009E000000	Image Description (0x010E)	ASCII (0x0002)	20 (0x00000014)	158 bytes (0x0000009E)
	0x0F01020014000000B2000000	Make (0x010F)	ASCII (0x0002)	20	178 bytes (0x000000B2)
Sony Ericsson W580i	0x0F0102000E00000086000000	Make (0x010F)	ASCII (0x0002)	13 (0x0000000E)	134 bytes (0x00000086)
	0x1001020006000000A6000000	Model (0x0110)	ASCII (0x0002)	6 (0x00000006)	166 bytes (0x000000A6)

Por tanto, para obtener el valor de la etiqueta *Image Description* hay que ir al lugar que indica el desplazamiento debido a que su tamaño es mayor de 4 bytes. Como la longitud es "0x9E" con respecto al inicio de la cabecera TIFF, los datos de la etiqueta comienzan en la posición "0xAA". A partir de este byte hay que contar 20 elementos de tipo ASCII (7-bit ASCII) por lo que el valor de la etiqueta es "SAMSUNG (12 espacios en blanco - "0x00")", terminando en NULL ("0x00") como se indica en Exif.

Al examinar la siguiente etiqueta del directorio 0th IFD para el mismo archivo se observa que la segunda entrada del directorio es "0x0F01020014000000B2000000", cuya interpretación se muestra también en la Tabla 3.6.

Por tanto, para obtener el valor de la etiqueta *Make* hay que ir al lugar que indica el desplazamiento. Como la longitud es "0xB2" con respecto al inicio de la cabecera TIFF, los datos de la etiqueta comienzan en la posición "0xBE". A partir de este byte hay que contar 20 elementos de tipo ASCII, por lo que el valor de la etiqueta es "SAMSUNG (12 espacios en blanco - "0x00")" terminando en NULL ("0x00") como se indica en Exif. Como puede verse dos etiquetas diferentes *Image Description* y *Make* pueden tener los mismos valores para una misma imagen; eso sí, su información debe ser duplicada para que siga la especificación Exif.

A continuación se van a analizar los mismos elementos del IFD para la imagen del Sony Ericsson W580i. Tras la cabecera TIFF están los bytes "0x0A00". Teniendo en cuenta que la alineación es "II" ("Intel"), estos bytes "0x0A00" indican cuantas entradas tiene el directorio actual, en este caso el 0th IFD. Por tanto, el 0th IFD tiene 10 entradas. La primera entrada del directorio "0x0F0102000E00000086000000" se interpreta como se muestra en la Tabla 3.6.

Por tanto, para obtener el valor de la etiqueta *Make* hay que ir al lugar que indica el desplazamiento. Como la longitud es "0x86" con respecto al inicio de la cabecera TIFF, los datos de la etiqueta comienzan en la posición "0x92". A partir de este byte hay que contar 13 elementos de tipo ASCII por lo que el valor de la etiqueta es "Sony Ericsson0x00" terminando en NULL ("0x00") como indica la especificación de Exif.

Al examinar la siguiente etiqueta del directorio 0th IFD se observa que la segunda entrada del directorio es "0x10010200060000000A6000000", cuyo significado se muestra también en la Tabla 3.6.

Por tanto, para obtener el valor de la etiqueta *Model* hay que ir al lugar que indica el desplazamiento. Como la longitud es "0xA6" con respecto al inicio de la cabecera TIFF, los datos de la etiqueta comienzan en la posición "0xB2". A partir de este byte hay que contar 6 elementos de tipo ASCII, por lo que el valor de la etiqueta es "W580i0x00" terminando en NULL ("0x00"), siguiendo la especificación Exif.

3.8 Síntesis del Capítulo

El objetivo de este capítulo ha sido describir los distintos tipos de metadatos en imágenes digitales, centrándose especialmente en la especificación Exif, ya que es la más utilizada por los fabricantes de dispositivos móviles. Se ha comenzado realizando una descripción detallada de la estructura y elementos que conforman Exif. A continuación se ha descrito brevemente los formatos de metadatos TIFF, JFIF, IPTC y XMP, que apenas son utilizados por las cámaras digitales. Finalmente, se ha realizado un análisis binario manual de los metadatos Exif de distintos ejemplos de imágenes. Este análisis manual es lento y tedioso y hace ver la necesidad de herramientas para su tratamiento automático.

Capítulo 4

Análisis Forense de Imágenes Digitales en Dispositivos Móviles

Este capítulo tiene como objetivo introducir las razones que justifican la necesidad del análisis forense de dispositivos móviles y realizar una descripción panorámica de los distintos tipos de técnicas forenses para la identificación de la fuente de adquisición de imágenes digitales. Primeramente se comienza describiendo las principales características que hacen a los dispositivos móviles fuentes potenciales del análisis forense. Seguidamente se exponen las distintas ramas del análisis forense centrándose en las imágenes digitales. Posteriormente se exponen los distintos conjuntos de técnicas forenses para la identificación de la fuente de adquisición de imágenes digitales. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

4.1 Necesidad del Análisis Forense en Dispositivos Móviles

Como se comentó en el primer Capítulo los dispositivos móviles proliferan a un ritmo imparable en nuestra sociedad. Los avances en las tecnologías de semiconductores permiten que dispositivos móviles amplíen a pasos agigantados su capacidad de procesamiento y almacenamiento. Esto, sumado a su uso cotidiano, hace que estos dispositivos sean frecuentemente evidencias en procesos judiciales o investigaciones policiales de todo tipo.

A continuación se van a describir casos en los que se aprecia de una forma clara y razonada la necesidad de herramientas para el análisis forense en este tipo de dispositivos [TNC10].

Un escenario de interés es la transmisión de información personal y corporativa. Las aplicaciones para los dispositivos móviles son desarrolladas en poco tiempo [AZ06]. Todo

tipo de aplicaciones de escritorio han sido portadas a dispositivos móviles. Los dispositivos móviles tienen capacidad para almacenar, ver e imprimir documentos electrónicos, transformándose en "oficinas móviles". De igual manera los dispositivos móviles son centro de envíos de mensajes (*Short Message Service* (SMS), *Multimedia Messaging System* (MMS), redes sociales, aplicaciones específicas) y correos electrónicos. Por tanto, hoy en día, los dispositivos móviles son fuente de adquisición, tratamiento y almacenamiento de información relevante de distinta naturaleza.

Otro escenario de interés es la utilización de los dispositivos móviles como centros de transacciones en línea: transacciones bancarias, compras en Internet, reservas de vuelos y hoteles, etc., donde se realizan operaciones con datos sensibles.

En el caso concreto del análisis forense de imágenes de dispositivos móviles no hay duda de la importancia que puede tener su aplicación en casos judiciales o criminales. Dado el gran uso de este tipo de cámaras y las polémicas que suscitan, son muchos los debates y normas que prohíben su utilización (empresas, conciertos, centros educativos, cenas y fiestas de negocios, ...). Asimismo, los dispositivos móviles pueden albergar fotografías almacenadas de carácter personal o fotografías que hayan sido tomadas in situ. Todo este conjunto de fotografías pueden ser evidencias de un hecho y elementos potenciales de uso en procesos judiciales y, consecuentemente, elementos de estudio del análisis forense.

4.2 Ramas del Análisis Forense de Imágenes Digitales

El análisis forense de imágenes digitales se puede dividir en dos grandes ramas [GKWB07]: autenticidad de imágenes digitales e identificación de la fuente de adquisición de una imagen.

La primera de las ramas trata de discernir si una imagen ha sufrido algún procesamiento posterior al de su creación, es decir, que no haya sido manipulada. La segunda de las ramas pretende identificar el tipo o clase de fuente que generó la imagen digital. Dentro de esta segunda rama puede realizarse una subdivisión en dos grupos: identificación del tipo de dispositivo fuente (cámara, escáner, generadas por computador, ...) o identificación de la marca y modelo del dispositivo.

Dentro de la identificación de la fuente de adquisición existen dos grandes enfoques: escenarios cerrados y escenarios abiertos.

Un escenario cerrado es aquel en el que la identificación de la fuente de la imagen se realiza sobre un conjunto de cámaras concreto y conocido a priori. Para este enfoque normalmente se utiliza un conjunto de imágenes de cada cámara para entrenar un clasificador y posteriormente se predice la fuente de adquisición de las imágenes objeto de

investigación. La técnica más utilizada para la tarea de clasificación de imágenes digitales es SVM [HCL03], aunque existen otras opciones como puede ser por ejemplo el uso de redes neuronales.

En los escenarios abiertos el analista forense no conoce a priori el conjunto de cámaras de las imágenes a identificar su fuente de adquisición. Obviamente en este tipo de clasificación, en la que no se tienen datos de cámaras a priori, el objetivo principal no es identificar la marca y modelo de la fuente, sino poder agrupar las distintas imágenes en conjuntos disjuntos en los que todos los elementos de cada conjunto pertenecen al mismo dispositivo. Este planteamiento es muy cercano a situaciones de la vida real, ya que en muchos casos el analista desconoce por completo el conjunto de cámaras a las que pueden pertenecer un conjunto de imágenes. Además, es prácticamente imposible tener un conjunto de imágenes de todos los dispositivos móviles existentes en el mundo para entrenar un clasificador. Aún así en el caso hipotético de poseer ese conjunto de imágenes el proceso de clasificación sería complejo y lento. Para el caso concreto de los escenarios abiertos es de gran utilidad poder agrupar las imágenes en conjuntos que pertenezcan a un mismo dispositivo, ya que esto puede aportar información muy válida y, en algunos casos, concluyente en las distintas investigaciones.

El éxito de las técnicas de identificación de la fuente de la imagen depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del mismo. Las características que se usan para identificar marca y modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan. Para el diseño de técnicas y algoritmos en cualquiera de estas ramas se aprovechan algunas características especiales de las imágenes que sirven como herramienta para el análisis forense. En [VCEK07] [TNC10] se realiza un estudio de las características que pueden ser objeto de análisis forense en dispositivos móviles. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes y que los algoritmos que usan para la generación de las imágenes también son muy similares entre modelos de la misma marca.

Según [VCEK07] se pueden establecer cuatro grupos de técnicas para este fin: basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en el uso de las características de la imagen y basadas en las imperfecciones del sensor. Asimismo, existe otro grupo de técnicas forenses a destacar: las basadas en los metadatos de la imagen.

4.3 Técnicas Basadas en Metadatos

Las cámaras digitales cuentan con una poderosa fuente de información que son los metadatos embebidos en los archivos de las imágenes digitales. Los metadatos o "datos sobre datos" registran información relacionada con las condiciones de captura de la imagen, como fecha y hora de generación, presencia o ausencia de *flash*, distancia de los objetos, tiempo de exposición, apertura del obturador e información GPS, entre otras. En otras palabras, información de interés que complementa el contenido principal de un documento digital. Los metadatos, entre otros usos, pueden llegar a ser una potente ayuda para la organización y búsqueda en librerías de imágenes.

Las imágenes digitales son almacenadas en una gran variedad de formatos como TIFF [Ado92], JPEG [Ham] o PSD. Algunos de los distintos contenedores de metadatos para los distintos formatos son: IFD Exif, TIFF, Adobe XMP [XMP13] e IPTC-IIM [ITPI07]. La especificación Exif [Com13] es la más utilizada para identificación de la fuente por ser el contenedor de metadatos más común en las cámaras digitales [Bae10]. La especificación Exif incluye cientos de etiquetas, entre las que se encuentran *marca* y *modelo*, aunque cabe destacar que la propia especificación no hace obligatoria su existencia en los archivos.

Sin embargo, estas técnicas dependen en gran medida de los metadatos que los fabricantes deciden insertar cuando la imagen es generada y la corrección en seguimiento de la especificación o estándar de metadatos que utilice. [SOAGGVHC12b] [SOAGGVHC14] realizan un estudio a fondo, donde se demuestra que los fabricantes no siguen fielmente la especificación Exif. Esto puede conllevar la extracción de información errónea o inválida para fines forenses. Asimismo, este método es el más vulnerable a modificaciones malintencionadas, e incluso se puede dar el caso de la eliminación total de los metadatos, ya sea intencionadamente o de manera inconsciente. Ejemplos de ello son algunos programas de edición fotográfica, que al editar o comprimir una imagen, actualizan incorrectamente los metadatos o provocan la pérdida de los mismos.

A pesar de las debilidades de este tipo de técnicas, si existen en el archivo los metadatos y de alguna manera se logra comprobar que no han sufrido modificaciones externas, su uso es de gran utilidad para los analistas forenses. Existe información difícilmente inferible del propio contenido de la imagen como por ejemplo la información GPS o la fecha y hora de la toma de la imagen, entre muchas otras.

4.4 Técnicas Basadas en la Aberración de las Lentes

Durante el proceso de generación de la imagen se pueden introducir aberraciones en la parte del sistema de lentes. Existen diferentes tipos de aberraciones: esférica, coma, astigmatismo, curvatura de campo, distorsión radial y distorsión cromática. La distorsión radial es la que más consecuencias tiene sobre la imagen, especialmente en las cámaras que usan lentes baratas de gran angular (*wide angle*). La mayoría de cámaras digitales usan este tipo de lentes por cuestiones de coste.

4.5 Técnicas Basadas en la Interpolación de la Matriz CFA

Algunos autores consideran que la elección de la matriz CFA y la especificación de los algoritmos de interpolación cromática generan algunas de las diferencias más marcadas entre los diferentes modelos de cámaras [BSM06] [CAS+06] [LH06] [BSM08]. Como se ha comentado en el Capítulo 2, las cámaras comerciales tienen un solo sensor en lugar de varios sensores para cada componente del color. En esencia, la interpolación cromática introduce un tipo específico de correlación entre los valores de colores de los píxeles de la imagen. La forma específica de estas dependencias se puede extraer de las imágenes para diferenciar los algoritmos de interpolación cromática y así determinar marca y modelo de la cámara que generó una imagen.

4.6 Técnicas Basadas en las Características de las Imágenes

Estas técnicas utilizan un conjunto de características extraídas del contenido de la escena de la imagen para realizar la identificación de la fuente. Existen tres grandes grupos de características clasificadas por su tipología: características de color, métricas de calidad de la imagen (*Image Quality Metrics* (IQM)) y estadísticas del dominio wavelet.

4.7 Técnicas Basadas en las Imperfecciones del Sensor

Estas técnicas se basan en el estudio de las huellas que los defectos del sensor pueden dejar sobre las imágenes. Estas técnicas se dividen en dos ramas: defectos de píxel y patrón del ruido del sensor (*Sensor Pattern Noise* (SPN)). En la primera se estudian los defectos de píxel, píxeles calientes, píxeles muertos, defectos de fila o columna y los defectos de grupo. En la segunda se construye un patrón del ruido promediando los múltiples residuos de ruido obtenidos mediante algún filtro de eliminación de ruido. La presencia del patrón se determina utilizando algún método de clasificación como correlación o máquinas SVM.

4.8 Síntesis del Capítulo

El objetivo de este capítulo ha sido introducir el análisis forense de dispositivos móviles, haciendo hincapié en la identificación de la fuente de adquisición de imágenes digitales. Se ha comenzado con la exposición de las razones que justifican la existencia y estudio de este tipo de análisis. Seguidamente se han descrito las principales ramas y situaciones existentes en el análisis forense de imágenes digitales. Finalmente, se ha mostrado una visión general de los distintos tipos de técnicas.

Capítulo 5

Trabajos Relacionados

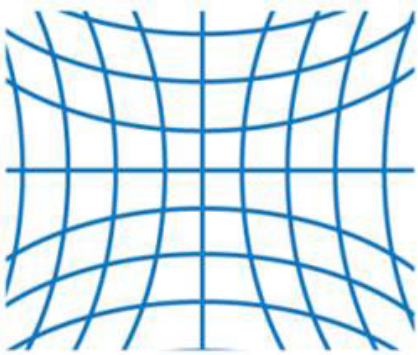
Este capítulo describe el estado del arte de las principales técnicas de análisis forense para la identificación de la fuente de adquisición de las imágenes digitales. Comienza con las técnicas de identificación basadas en los metadatos. Seguidamente se exponen las referencias más relevantes sobre los distintos tipos de técnicas establecidos en [VCEK07]: basadas en la aberración de las lentes, basadas en la interpolación de la matriz CFA, basadas en las características de las imágenes y basadas en el uso de las imperfecciones del sensor. Posteriormente, se presenta un cuadro resumen del estado del arte de las técnicas de identificación de la fuente de adquisición presentadas. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo y se presenta un cuadro resumen del estado del arte de las técnicas de identificación de la fuente de adquisición presentadas. Cabe destacar que aunque este trabajo esté centrado en los dispositivos móviles, en el estado del arte se aúnan referencias de técnicas sobre imágenes de todo tipo de dispositivos, ya que su conocimiento puede ser válido para la aplicación o adaptación a imágenes de dispositivos móviles.

5.1 Técnicas Basadas en Metadatos

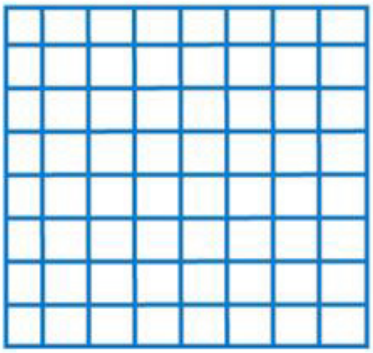
Las técnicas basadas en el análisis de los metadatos de la imagen son las más sencillas. Existen gran cantidad de trabajos sobre los diferentes tipos de metadatos, tanto para la búsqueda de información, como para la clasificación de imágenes [Pla00] [BL05] [Tes05] [RCC+08] [AG11]. Asimismo, los metadatos pueden utilizarse como datos de entrada o ayuda para el uso de otras técnicas forenses. Por ejemplo, en la aplicación de técnicas basadas en el contenido de la imagen, los metadatos Exif puede ofrecer una gran cantidad y variedad de información de aspectos técnicos, que pueden permitir aumentar las tasas de acierto o mejorar los resultados de la aplicación de ciertos algoritmos forenses [BL04] [CJJYJwHG07] [JKCS11].

5.2 Técnicas Basadas en la Aberración de las Lentes

En [Cho06] se propone la distorsión radial de la lente como técnica para la identificación de la fuente. Los autores concluyen que cada modelo de cámara expresa un único patrón de distorsión radial que ayuda a identificarla de manera única. En los experimentos se utilizan tres cámaras diferentes y se obtiene como resultado una precisión del 87 al 91 % en la identificación de la fuente. La Figura 5.1 presenta un ejemplo del efecto de la distorsión radial.



(a) Imagen con distorsión radial negativa



(b) Imagen sin distorsión

(c) Imagen con distorsión radial positiva

Figura 5.1: Ejemplos de efecto de la distorsión radial

En [LTEK07] se propone la aberración cromática lateral como técnica para la identificación de la fuente. Los autores realizan distintos tipos de experimentos utilizando conjuntos de cámaras con imágenes no modificadas, modificadas o con recortes aleatorios en regiones de la imagen. En el experimento en el que se usan tres cámaras de diferentes marcas se obtiene una precisión del 86,67 % en la identificación de la fuente. Asimismo, se concluye que esta técnica no es adecuada para la identificación de la fuente de distintos modelos de cámara de la misma marca.

5.3 Técnicas Basadas en la Interpolación de la Matriz CFA

En [LH06] se utilizan las correlaciones entre píxeles en el proceso de identificación de la fuente. Definen un modelo de correlación cuadrática de los píxeles y obtienen una matriz de coeficientes para cada banda de color. Para la clasificación utilizan redes neuronales. Se probó el método para cuatro cámaras y la tasa de acierto osciló entre el 95 % y el 100 %, con una tasa media del 98,25 %. También se realizan pruebas para imágenes modificadas (incluyendo compresión) con resultados de un 80 % de éxito para una compresión JPEG del 80 %. Dado que las cámaras del mismo fabricante utilizan el mismo algoritmo de interpolación cromática, esta técnica no es eficiente para diferenciar entre distintos modelos del mismo fabricante. Asimismo, como demuestran los experimentos, no se obtienen buenos resultados cuando las imágenes han sido modificadas o tienen un nivel alto de compresión.

En [CAS+06] se utiliza un conjunto de medidas de similitud binarias como métricas para estimar la semejanza entre los planos de bits de una imagen. El supuesto en el que se basa este trabajo es que el algoritmo de interpolación CFA de cada fabricante deja correlaciones a lo largo de los planos de bits de una imagen y éstas pueden ser representadas por este conjunto de medidas. En este estudio se utilizan 108 medidas de similitud binarias. Los experimentos realizados con esta técnica para clasificar 3 grupos de cámaras obtienen un porcentaje de éxito entre el 81 % y el 98 %, mientras que para un grupo de 9 cámaras la precisión desciende al 62 %. Claramente se puede apreciar que los resultados del método dependen del número de cámaras utilizadas en los experimentos.

En [BSM08] se presenta un algoritmo para identificar y clasificar las operaciones de interpolación cromática. La propuesta se basa en dos métodos para realizar el proceso de clasificación: el primer método utiliza un algoritmo de esperanza-maximización (*Expectation-Maximization* (EM)) para analizar la correlación del valor de cada píxel con los valores de sus vecinos y el segundo método realiza un análisis de las diferencias entre píxeles (*inter-píxel*). Se realizan diferentes experimentos con distintos números de cámaras y distintos tipos de imágenes. Los resultados obtenidos en la identificación de la fuente de una imagen varían entre el 84,8 % y el 92,56 % de tasa media de acierto.

En [CK10] se presenta una técnica para la identificación de la fuente basada en la información del proceso de interpolación de la matriz CFA y una comparativa con otras técnicas. Esta técnica presenta tres nuevos grupos de características de *demosaicing* distintas a las definidas en los grupos anteriores: *weights*, *Error Cumulants* (EC) y *Normalized Group Sizes* (NGS). Dado que el número de características es muy alto se realiza un proceso (*Eigenfeature Regularization* (ERE)) para disminuir el número de las mismas. Se realizan distintos experimentos utilizando clasificadores *First Nearest Neighbor* (1NN) y *Probabilistic Support Vector Machine* (PSVM). Los resultados utilizando 15 cámaras de 4 fabricantes distintos y 11 modelos diferentes (hay cámaras de la misma marca y del mismo

modelo), con una reducción a 20 características y un clasificador PSVM, obtienen unos resultados de acierto medio del 99,4 % para la distinción de la marca y un 94,8 % para la distinción del modelo.

En [HAJY10] se proponen cuatro algoritmos que utilizan aspectos basados en la correlación *inter-channel*. Estos algoritmos calculan mapas de varianza (*v-maps*) y los clasifican utilizando 1NN. En los experimentos para la identificación de la fuente de la imagen se utilizan cuatro cámaras de tres fabricantes distintos y 50 imágenes de cada una (25 para entrenar y 25 para el test). Los resultados muestran un acierto medio del 94,5 % y los autores concluyen que la correlación *inter-channel* ofrece un enfoque complementario a trabajos anteriores que tratan correlaciones entre los píxeles introducidas por el proceso de *demosaicing*.

5.4 Técnicas Basadas en las Características de las Imágenes

En [TLL07] se propone un método de identificación de la fuente utilizando las siguientes características: color, calidad de la imagen y dominio de la frecuencia. En el estudio adoptan la Transformada Wavelet como método para calcular las estadísticas del dominio wavelet y utilizan SVM para la clasificación. En los experimentos se usan cámaras digitales y dispositivos móviles. Los resultados obtenidos en los distintos experimentos arrojan unos resultados entre el 61,7 % y el 99,72 % de acierto.

En [MSGW08] se extiende la identificación de la fuente a diferentes dispositivos tales como teléfonos móviles con cámara integrada, cámaras digitales, escáneres y computadoras. En esta propuesta se usan las diferencias en el proceso de adquisición de la imagen de los dispositivos para formar dos grupos de características: coeficientes de interpolación de color y características de ruido. En los experimentos se utilizan cinco modelos de teléfonos móviles, cinco modelos de cámaras digitales y cuatro modelos de escáneres para identificar el tipo de fuente. En los resultados globales se obtiene un 93,75 % de precisión. En el análisis de identificación de marca y modelo de teléfonos móviles obtienen una precisión del 97,7 % para los cinco modelos.

En [WGKM09] se propone un método para la identificación de la cámara fuente mediante la extracción y clasificación de las estadísticas de las características *wavelets*. Este método está compuesto por tres fases: extracción, selección y clasificación de características wavelet. Las características del dominio wavelet se extraen para integrar un modelo estadístico de imagen a partir de los coeficientes wavelet, incluyendo 216 características wavelet de primer orden y 135 características de co-ocurrencia de segundo orden. En este estudio las características del dominio wavelet se consideran más representativas y se prefieren a las características espaciales (color de la imagen e IQM) y matrices de filtros

de color CFA. De manera análoga a [MKY08] se realiza una descomposición wavelet en 4 niveles basada en *Quadrature Mirror Filter* (QMF) para dividir el espacio de la frecuencia, se extraen las mismas cuatro estadísticas (media, varianza, asimetría y curtosis) junto a los errores de predicción lineal. Ya que estas cuatro estadísticas no brindan información sobre la correlación de la textura se usan las características de co-ocurrencia para la extracción de características de textura de la imagen, ya que según [RH99] son las más adecuadas para este propósito. A partir de las características de co-ocurrencia se extraen las características de segundo orden (energía, entropía, contraste, homogeneidad y correlación). Por último, y al igual que en [MKY08], se seleccionan las características más representativas utilizando un algoritmo *Sequential Floating Forward Selection* (SFFS) y se clasifican utilizando SVM. Bajo las mismas condiciones que en los experimentos realizados en [MKY08] se consigue una media de identificación del 98 % para el conjunto de todas las cámaras y una tasa media de acierto del 96,9 % para las tres cámaras del mismo modelo.

En [HLZ10] se realizan experimentos con las características más usuales de las imágenes para la identificación de la fuente: wavelet, color, IQM, características estadísticas de la diferencia entre imágenes y características estadísticas de predicción de errores. En los experimentos se proponen distintas combinaciones de los distintos tipos de características y SVM para la clasificación de los distintos dispositivos. Se utilizan diez cámaras diferentes de cuatro fabricantes distintos con 300 imágenes de cada cámara (150 para entrenamiento y 150 para predecir) y de una resolución de 1024×1024. Utilizando todas las características se obtiene un resultado de acierto medio del 92 %. Asimismo, se realizan experimentos para comprobar la robustez ante tres de las manipulaciones más comunes en imágenes digitales: la compresión JPEG, el recortado y el escalado. Las conclusiones finales de este trabajo son que algunos de los conjuntos de características ofrecen buenas tasas de aciertos para imágenes intactas, pero no para las que tienen modificaciones. También se muestra que diferentes tipos de manipulaciones tienen efectos diferentes sobre las tasas de acierto de los diferentes conjuntos de características.

En [OA11] se plantea una técnica para, entre otros fines, la identificación de la fuente de una imagen utilizando los modelos estadísticos para *ridgelet* y sub-bandas *contourlet*. Tras la extracción de las características se aplica un algoritmo SFFS para selección de características y SVM para la clasificación. El método basado en *wavelets* considera 216 características útiles sólo para la representación de una dimensión, el enfoque basado en *ridgelets* toma 48 características y la aproximación de *contourlets* contempla un total de 768 características. En los experimentos realizados con tres cámaras de distintos fabricantes se obtienen porcentajes de acierto entre el 99,5 % y el 99,8 %. Los *contourlets* y *ridgelets* no sólo son efectivos para diferenciar entre modelos de cámaras, sino también para diferenciar entre imágenes naturales o producidas por ordenador, o para diferenciar imágenes de escáneres de la misma marca. De cualquier manera los autores consideran que podrían implementar mejoras experimentando con diferentes algoritmos de selección.

En [LLC+12] se propone un método que emplea la densidad marginal de los coeficientes de la *Discrete Cosine Transform* (DCT) en las coordenadas de baja frecuencia y las características de densidad conjunta de vecindad en el domino DCT. Adicionalmente, se utiliza la agrupación jerárquica y SVM para detectar la fuente de adquisición de las imágenes. En los experimentos realizados con imágenes pertenecientes a cinco modelos de teléfonos inteligentes de cuatro fabricantes, se obtiene entre el 86,36 % y el 99,91 % de acierto, alcanzando los mejores resultados con un *kernel* SVM lineal.

5.5 Técnicas Basadas en las Imperfecciones del Sensor

En [GBK+01] se estudian los defectos de los píxeles en los sensores de tipo CCD, centrándose en la evaluación de diferentes características para examinar las imágenes e identificar la fuente: defectos del sensor CCD, formato de los archivos usados, ruido introducido en la imagen y marcas de agua introducidas por el fabricante de la cámara. Entre los defectos del sensor CCD considerados se encuentran los puntos calientes, los píxeles muertos, los defectos en grupo y los defectos de fila o columna. En los resultados se observa que cada una de las cámaras tiene un patrón de defecto diferente. Sin embargo, también se señala que el número de defectos en los píxeles para una cámara difiere de una foto a otra, oscilando ampliamente en función del contenido de la imagen. Asimismo, se revela que el número de defectos cambia con la temperatura. Por último, el estudio encuentra que las cámaras con CCD de alta calidad no tienen este tipo de problema. También es cierto que la mayoría de las cámaras tienen mecanismos adicionales para compensar este tipo de problemas. Al considerar únicamente los defectos de los sensores de tipo CCD este estudio no es aplicable al análisis de imágenes generadas por dispositivos móviles.

En [LFG06] se analiza el patrón del ruido del sensor de un conjunto de cámaras, el cual funciona como una huella dactilar, permitiendo la identificación única de cada cámara. Para calcular este patrón se realiza un promedio del ruido obtenido en diferentes imágenes utilizando un filtro de eliminación de ruido. Para identificar la cámara a partir de una imagen dada, se considera el patrón de referencia como una marca de agua cuya presencia en la imagen es establecida mediante un detector de correlación. El estudio utiliza aproximadamente 320 imágenes procedentes de 9 cámaras (2 son exactamente del mismo modelo), obteniéndose buenos resultados. Cabe destacar que este porcentaje de éxito se debe a que en los experimentos los autores utilizan el mismo conjunto de imágenes para calcular el patrón de referencia y las correlaciones. También se demuestra que este método está afectado por algoritmos de procesamiento de la imagen, como la compresión JPEG y la corrección *gamma*. Según [VCEK07] los resultados para fotografías recortadas no son satisfactorios. Asimismo, en esta técnica las imágenes de las que se extrae el patrón de referencia tienen que tener el mismo tamaño que las imágenes a examinar.

En [CESR12] se propone un enfoque para la identificación fuente de la cámara considerando escenarios abiertos, donde, a diferencia de los escenarios cerrados, no se da por sentado contar con acceso a todas las posibles cámaras de origen de la imagen. Esta propuesta comprende tres fases: definición de las regiones de interés, determinación de las características e identificación de la cámara fuente. Las diferentes regiones de las imágenes pueden contener información distinta sobre la huella digital de la cámara fuente. Este enfoque, en contraste con otros, considera nueve áreas de interés (*Region of Interest* (ROI)) y no sólo la región central de la imagen. La Figura 5.2 muestra las 9 áreas de interés consideradas.

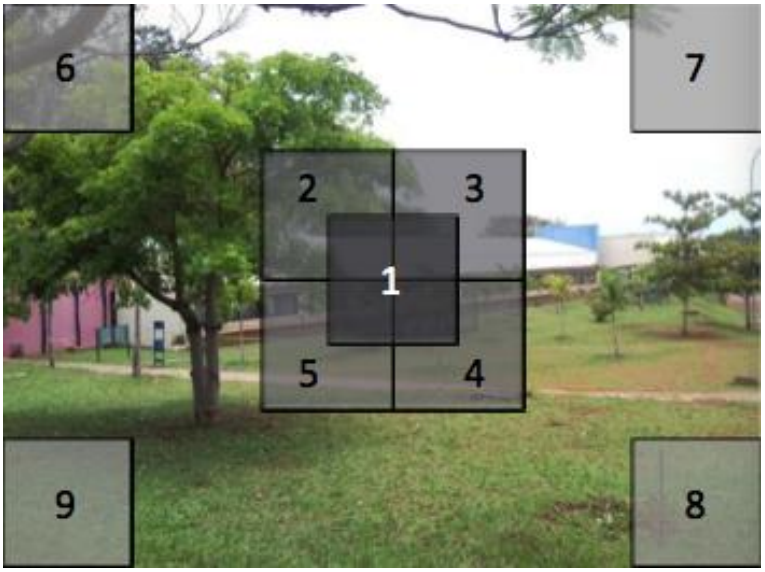


Figura 5.2: 9 áreas de interés

El uso de las regiones de interés facilita trabajar con imágenes de diferentes resoluciones. Para determinar las características se calcula el SPN para cada uno de los canales R, G y B. Asimismo, se calcula el SPN para el canal Y (luminancia), que es una combinación de los tres canales *Red-Green-Blue* (RGB) (similar a una versión en escala de grises de la imagen), generándose un total de 36 características para representar cada imagen. Después, las imágenes tomadas por la cámara bajo investigación son etiquetadas como clases positivas y las tomadas por las cámaras disponibles restantes como clases negativas. En la fase de entrenamiento de SVM se calcula el hiper-plano que separa los casos positivos y negativos. Posteriormente, se tienen en cuenta las clases desconocidas del escenario abierto, moviendo el hiper-plano generado hacia adentro (hacia las clases positivas) o hacia afuera (hacia las clases negativas). Mediante el movimiento del hiper-plano se puede variar el margen para determinar si una imagen pertenece a una clase u otra. A este proceso se le denomina modelado de límites de decisión. En los experimentos se utiliza un conjunto

de 25 cámaras digitales de 9 fabricantes, 150 imágenes de cada cámara en formato JPEG con diferentes configuraciones de luz, *zoom* y *flash*. Los resultados de los experimentos muestran una precisión del 94,49 %, del 96,77 % y del 98,10 %, utilizando conjuntos abiertos con 2/25, 5/25 y 15/25 cámaras, respectivamente, definiendo un conjunto abierto x/y como el conjunto de y cámaras donde x cámaras son conocidas y utilizadas para entrenar e $y - x$ son las cámaras desconocidas, cuyas imágenes junto con las de las de las cámaras conocidas son utilizadas en la fase de test.

En [dOCSE+14] se realiza una extensión del artículo [CESR12], donde además de presentar otras técnicas y algoritmos, se realizan nuevos experimentos. En los experimentos se utilizan 13210 imágenes de 400 cámaras (sólo se tenía acceso físico a 25 cámaras, el resto son imágenes descargadas de *Flickr*) y los mejores resultados obtienen tasas de acierto del 96,56 %, 97,34 %, 96,80 % y del 97,18 %, utilizando conjuntos abiertos con 2/25, 5/25, 10/25 y 15/25 cámaras, respectivamente.

5.6 Síntesis del Capítulo

El objetivo de este capítulo ha sido presentar los trabajos más representativos de las técnicas forenses de identificación de la fuente de adquisición de imágenes digitales. Se ha comenzado con la exposición de los distintos tipos de técnicas de identificación de la fuente de adquisición de imágenes digitales. Otros compendios de técnicas pueden verse en [SWL09] [RSBG11] [SORCAG+13]. Finalmente, se ha presentado una tabla resumen que facilita una visión general de las diferentes técnicas.

Como se ha visto, la mayoría de los trabajos relacionados no se han realizado con imágenes de dispositivos móviles. Este punto es importante a tener en cuenta, ya que muchas de las técnicas, dada la distinta naturaleza física de las cámaras de los dispositivos móviles, no son válidas o necesitan de adaptaciones para su aplicación a imágenes de dispositivos móviles.

En la Tabla 5.1 se muestra un resumen del estado del arte descrito anteriormente. La información no detallada en los artículos correspondientes ha sido cumplimentada con las siglas ND (No Detallado). Hay que tener en cuenta que en la mayoría de los artículos anteriores se realizan diferentes experimentos con distinto número de cámaras e imágenes. En la columna "Número de Modelos/Marcas" se contabilizan el total de modelos y fabricantes utilizados para todos los experimentos, lo cual no implica que en todos los experimentos se utilicen todos los modelos de todos los fabricantes. La columna "Aplicado a Dispositivos Móviles" indica que al menos uno de los modelos utilizados en los experimentos es un dispositivo móvil. La columna "Aplicado a Diferentes Modelos de la Misma Marca" indica que en al menos uno de los experimentos se han utilizado cámaras del mismo fabricante.

En cada experimento se obtiene una tasa media de acierto en la identificación de la fuente. Las "Tasas Mínima y Máxima de Acierto" muestran, respectivamente, los valores mínimos y máximos de acierto en los distintos experimentos realizados. En el caso de que haya un único valor es porque en ese artículo solamente se ha realizado un experimento.

Tabla 5.1: Comparativa sobre las diferentes técnicas de identificación de la fuente de adquisición de una imagen

Técnica	**Propuesta**	**Método de Clasificación**	**Número de Modelos/Marcas**	**Formato de las Imágenes**	**Resolución**	**Aplicado a Dispositivos Móviles**	**Aplicado a Diferentes Modelos de la Misma Marca**	**Tasas Mínima y Máxima de Acierto**
Aberración de las lentes	[Cho06]	SVM	3/3	JPEG	Diferentes	ND	No	87,38 % - 91,53 %
	[LTEK07]	SVM	3/3	JPEG	Diferentes	No	Sí	72,75 % - 92,22 %
Interpolación matriz CFA	[LH06]	Red Neuronal	4/4	Sin compresión	ND	No	No	98,25 %
	[CAS+06]	SVM Lineal y No Lineal RBF	9/3	ND	Diferentes	Sí	Sí	62,3 % - 98,7 %
	[BSM08]	SVM Lineal y No Lineal RBF	5/5	JPEG	Diferentes	No	Sí	84,8 % - 88 %
	[CK10]	PSVM y 1NN	4/11	JPEG	Diferentes	Sí	Sí	94,8 % - 99,4 %
	[HAJY10]	1NN	4/3	JPEG	ND	No	Sí	94,5 %
Características de las imágenes	[TLL07]	SVM Lineal	2/7	JPEG	1600×1200	Sí	Sí	61,7 % - 99,72 %
	[MSGW08]	SVM Lineal	5/5	JPEG	ND	Sí	No	97,7 %
	[WGKM09]	SVM No Lineal RBF	6/4	JPEG	ND	No	Sí	98 %
	[HLZ10]	SVM No Lineal RBF	10/4	JPEG	1024×1024	No	Sí	47 % - 92 %
	[OA11]	SVM No Lineal RBF	3/3	ND	ND	No	No	93,33 % - 99,7 %
	[LLC+12]	SVM Lineal y No Lineal RBF	5/4	JPEG	Diferentes	Sí	Sí	86,36 % - 99,91 %
Imperfecciones del sensor	[GBK+01]	ND	2/2	ND	640×480	No	No	ND
	[LFG06]	ND	5/9	Diferentes	Diferentes	No	Sí	ND
	[CESR12]	SVM No Lineal RBF	25/9	JPEG	Diferentes	Sí	Sí	94,49 % - 98,10 %
	[dOCSE+14]	SVM No Lineal RBF	25/9	JPEG	Diferentes	Sí	Sí	96,56 % - 97,34 %

Capítulo 6

Theia: Herramienta para el Análisis Forense de Imágenes

Este capítulo presenta *Theia*, una herramienta para el análisis forense de imágenes digitales. El capítulo comienza describiendo las principales características de la herramienta y sus funcionalidades con respecto al análisis de metadatos. Seguidamente se muestran los principales componentes y la estructura del diseño e implementación de la herramienta. Posteriormente se realiza una comparativa con las principales herramientas existentes para el mismo fin. Finalmente se presentan ejemplos concretos de anomalías o errores en el tratamiento de metadatos Exif por parte de las herramientas con las que se compara *Theia*. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

6.1 Generalidades

Claramente puede entenderse que la obtención de los metadatos a nivel binario manualmente es tediosa y lenta. Por tanto, se necesitan herramientas para la extracción automática y su visualización de forma gráfica y amigable. Este tipo de herramientas sirven de apoyo a la tarea del analista forense en lo que respecta al análisis de los metadatos. Sin el uso de este tipo de aplicaciones sería complejo realizar el procesamiento de un gran número de imágenes. Asimismo, no es simplemente un tema de ayuda al analista forense con respecto al procesamiento de gran número de imágenes o la optimización del tiempo, aporta fiabilidad al proceso y ofrece diferentes funcionalidades complementarias.

En este trabajo se ha desarrollado una herramienta denominada *Theia* que facilita la extracción y el tratamiento de metadatos Exif en imágenes JPEG. A grandes rasgos la herramienta se divide en dos grandes partes:

- **Tratamiento de imágenes a nivel individual**: Permite obtener la información Exif detallada de una imagen individual, encontrar modificaciones realizadas en la imagen al compararla con la imagen en miniatura existente en la información Exif y situar la imagen en Google Maps y Google Earth (si posee información de geoposicionamiento), como se puede observar en las Figuras 6.1 y 6.2. A la hora de mostrar la información Exif se ha organizado en 6 grupos: *Image*, *Exif*, *GPS*, *Interoperability*, *Thumbnail* y *Maker Note*.

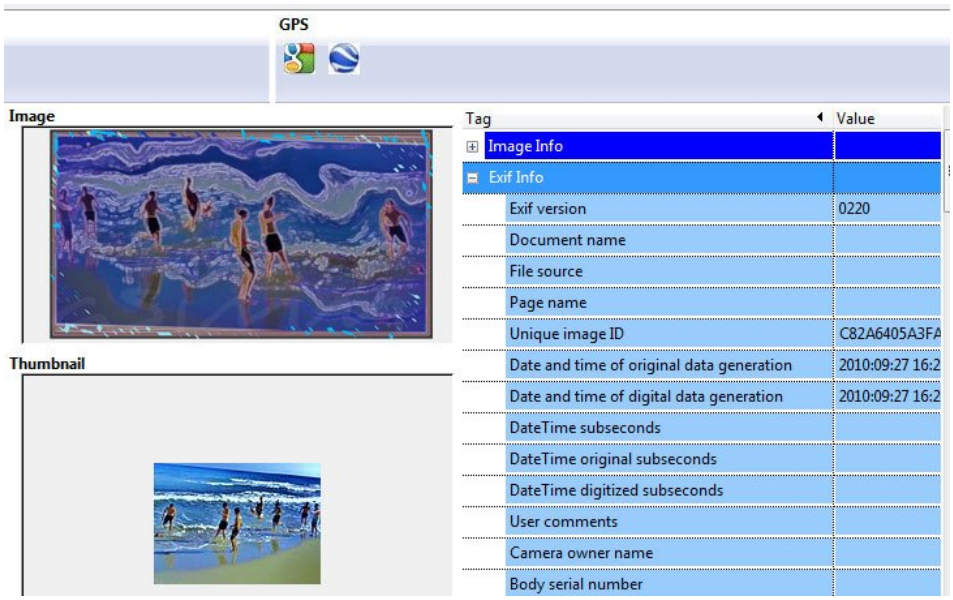


Figura 6.1: Análisis individual de una imagen

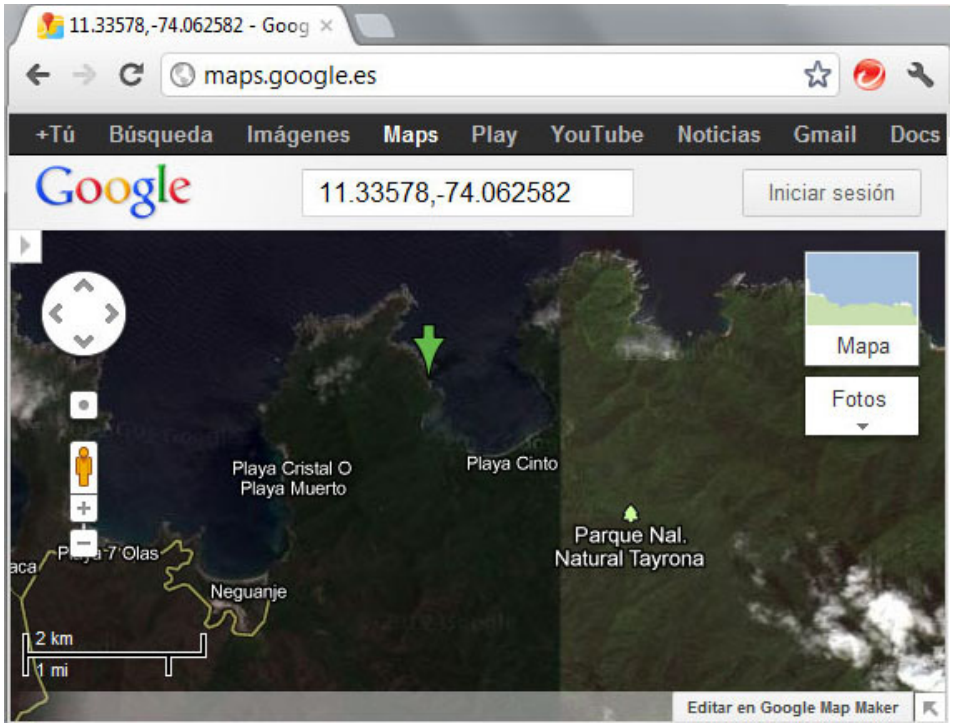


Figura 6.2: Geoposicionamiento de una imagen en Google Maps

- **Tratamiento de imágenes a nivel de grupo**: Permite hacer análisis de imágenes en grupo. Cada grupo es totalmente independiente entre sí. Los diferentes análisis que se pueden realizar sobre cada grupo son los siguientes: administración de imágenes (añadir y eliminar imágenes), consultas preestablecidas, análisis de modificaciones basadas en la imagen en miniatura almacenada, consultas avanzadas y geoposicionamiento de las imágenes.

  - *Consultas preestablecidas*: Permite crear consultas agregando etiquetas Exif (y otras adicionales que añade la aplicación que ayudan al análisis forense) sobre las imágenes del grupo seleccionado. La consulta agrupa las imágenes por los criterios seleccionados y muestra el número de imágenes que hay en cada uno de los grupos formados, como puede verse en un ejemplo en la Figura 6.3.
  - *Análisis de modificaciones basadas en la imagen en miniatura almacenada*: Tiene como objetivo determinar si se realizaron modificaciones en las imágenes del grupo posteriores a la captura de las mismas. Esta operación se realiza calculando el *Root Mean Square* (RMS) de la comparación de la imagen en miniatura que se encuentra en la información Exif con la imagen en miniatura generada a partir de la imagen analizada. Clasifica las imágenes en: sin modificaciones, posiblemente modificadas y con modificaciones, representadas con los colores verde, naranja y rojo, respectivamente, como se muestra en la Figura 6.4.

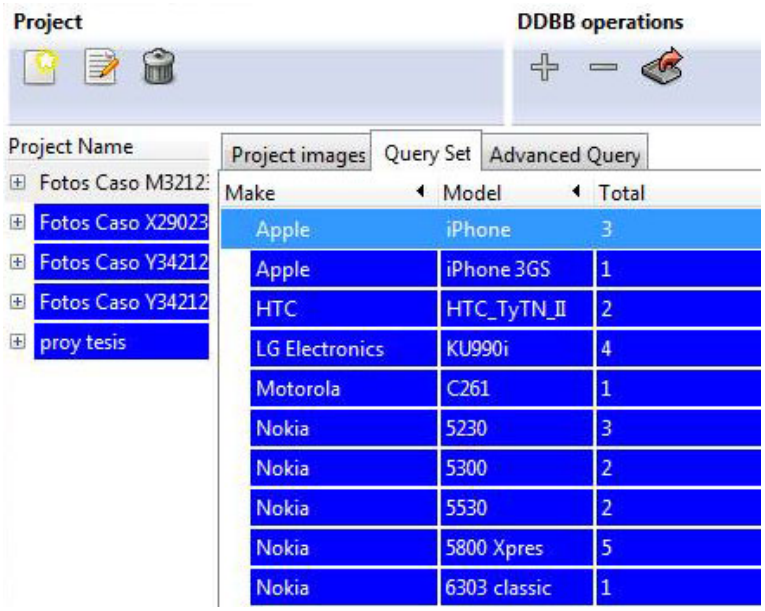


Figura 6.3: Consultas preestablecidas

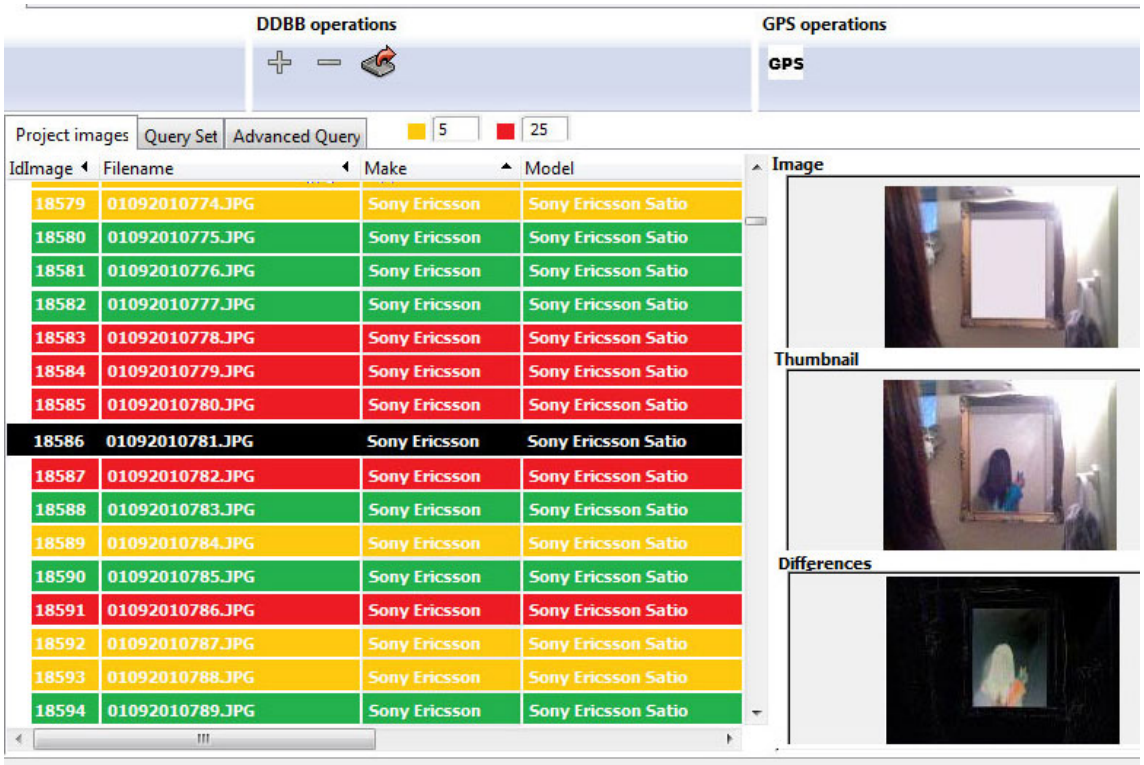


Figura 6.4: Análisis de la imagen en miniatura

– *Consultas avanzadas*: Permite la creación de consultas sobre imágenes de un grupo configurando los datos Exif a mostrar y los filtros a aplicar. Es decir, muestra la información de las imágenes de los campos seleccionados que coincidan con uno de los valores de cada uno de los filtros configurados. Asimismo, se permite el almacenamiento permanente de consultas. Una visión general se muestra en la Figura 6.5.

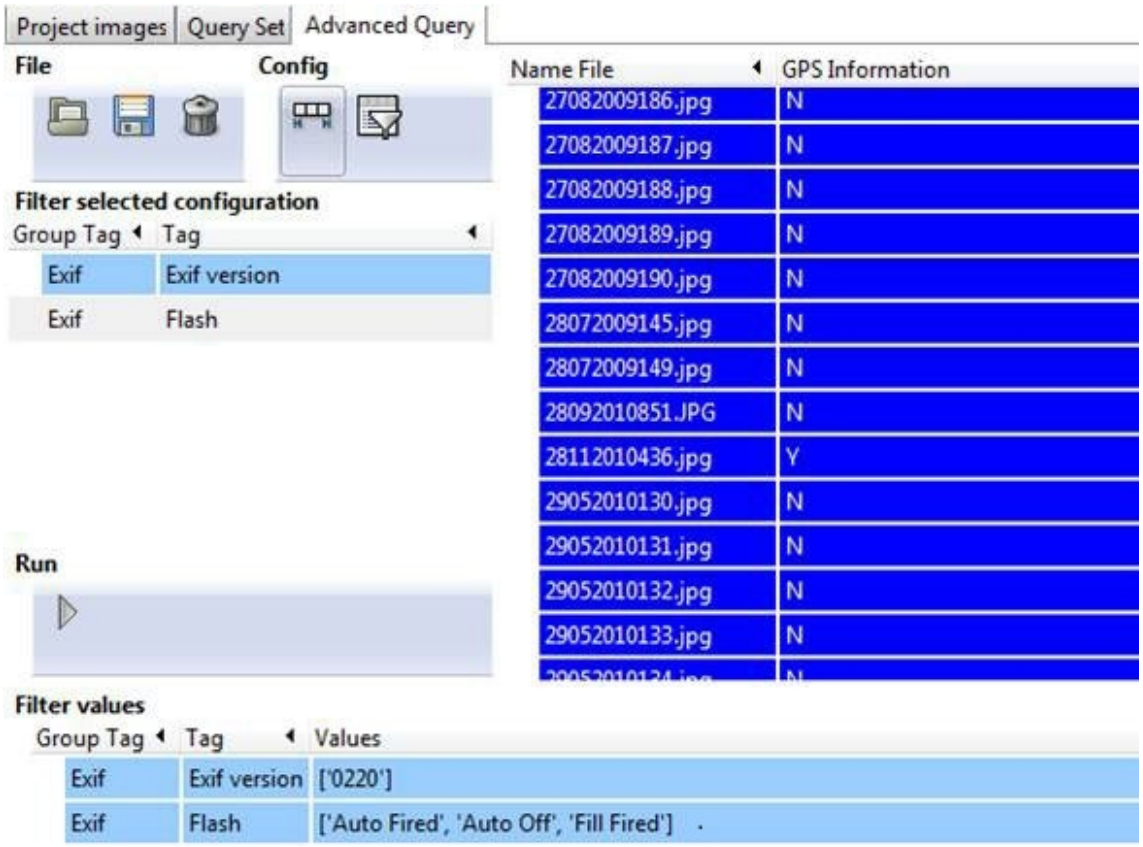
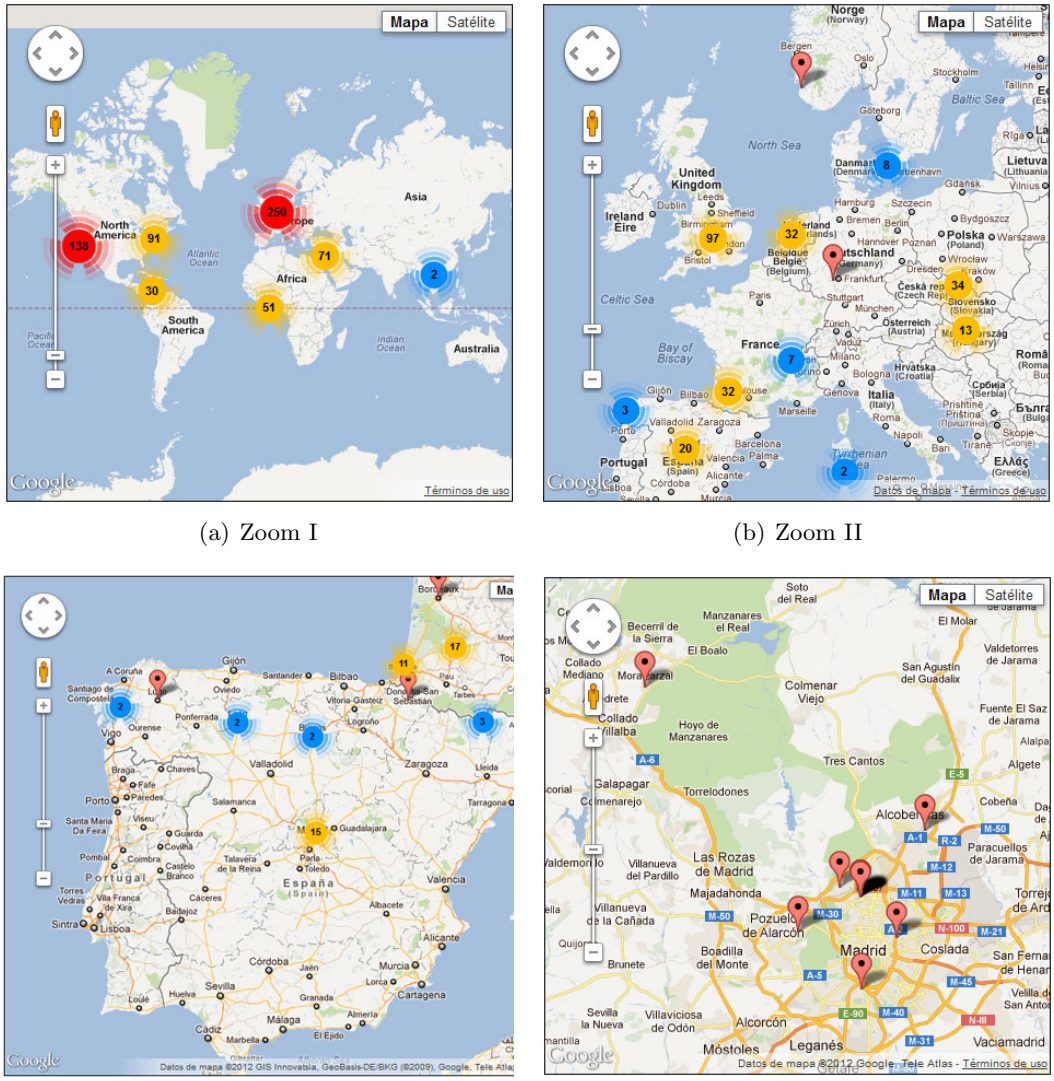


Figura 6.5: Consultas avanzadas

– *Geoposicionamiento*: Análogamente al tratamiento de imágenes a nivel individual, existe una funcionalidad que permite el tratamiento de la información de geoposicionamiento para un grupo de imágenes. Esta opción permite la selección de algunas o de todas las imágenes de un grupo con información de geoposicionamiento para la creación de un mapa en Google Maps que sitúe a las mismas. En el mapa se agrupan las imágenes por zona y, a medida que se aumenta el *zoom*, se van detallando las coordenadas. La Figura 6.6 muestra un ejemplo del mapa generado y el proceso de aumento del *zoom* en una zona concreta (desde la Figura 6.6(a) hasta la Figura 6.6(d)).



(a) Zoom I

(b) Zoom II

(c) Zoom III

(d) Zoom IV

Figura 6.6: Geoposicionamiento de un grupo de imágenes en Google Maps

6.2 Diseño

Como aspectos relevantes de la implementación pueden señalarse las diversas herramientas utilizadas para llevar a cabo la construcción de los distintos elementos que forman la aplicación:

- El lenguaje de programación utilizado para la codificación de la aplicación es Python 2.6. De éste se destaca la utilización de la librería *Python Imaging Library* (PIL) que facilita diversos tipos de tratamientos sobre imágenes. El código fuente está estructurado en 8 paquetes:

  - *Exif*: Es el corazón de la extracción de metadatos Exif de la aplicación. Se compone de clases y un conjunto de estructuras de datos auxiliares que permiten la obtención de forma eficiente de los metadatos Exif hasta la versión 2.3.
  - *BBDD*: En este paquete se encuentra todo el código relativo al control de la base de datos. Se compone de clases que permiten la conexión y la gestión de todos los datos de la aplicación, ya que tanto los datos de usuario como los de configuración están almacenados en la base de datos.
  - *Interface*: Se encarga de controlar el flujo principal de la aplicación y todo el interface de la misma.
  - *GPS*: Aporta toda la funcionalidad relacionada con el geoposicionamiento de las imágenes. Esta funcionalidad se apoya en los servicios de geoposicionamiento que Google ofrece por Internet.
  - *Exception*: Contiene el conjunto de clases para controlar de manera más adecuada las excepciones de la aplicación.
  - *Features*: Este paquete incluye el conjunto de clases que gestiona los algoritmos de identificación de la fuente de adquisición de imágenes basados en el contenido de la imagen.
  - *LibSVM*: Abarca el conjunto de clases que gestiona la configuración de los distintos parámetros de la máquina SVM. Esta máquina SVM se utiliza para la clasificación de las imágenes en los algoritmos de identificación de la fuente de adquisición.
  - *PRNU*: Este paquete contiene las clases que gestionan la extracción del patrón del ruido *Photo Response Non-Uniformity* (PRNU) de las imágenes.

- Se ha utilizado como plataforma de programación Eclipse Helios, ya que permite crear entornos integrados de desarrollo multilenguaje y adaptables. Ofrece una extensa flexibilidad de configuración con los complementos necesarios para adaptar los requerimientos de desarrollo que no estén contemplados dentro de las configuraciones básicas. Para este caso concreto se requirió el uso del complemento Pydev de Eclipse para permitir el desarrollo en Python.

- Para el diseño de cada unos de los formularios de las ventanas de la aplicación se ha utilizado Glade 3. Esta herramienta de desarrollo visual de interfaces gráficas se basa en el uso de las librerías gráficas de Gtk/Gnome (en este caso pygtk) y genera ficheros con extensión ".glade".

- Para la lectura y posterior ejecución de las interfaces de los ficheros en formato ".glade" por código Python se ha utilizado el paquete Tepache.py. Tepache crea automáticamente un esquema de la clase en código Python que permite el control y ejecución de la interfaz generada por Glade 3.

- Para la realización de la documentación interna del proyecto se ha utilizado Doxygen.
- La gestión de versiones de todos los archivos generados en el proyecto se realiza con Subversion.
- El motor de base de datos utilizado es MySQL. En ella se almacena toda la información necesaria para el funcionamiento de la aplicación. Las tablas que conforman la base de datos se han organizado en 4 grupos: tablas de configuración, tablas de generación de consultas, tablas de binarios de las imágenes y tablas de información Exif.

6.3 Comparativa

Para realizar una comparativa de *Theia* con otras con fines similares se han buscado principalmente herramientas de extracción y tratamiento de metadatos Exif para archivos JPEG, aunque no ha sido un criterio que excluya a otro tipo de herramientas relacionadas.

Las herramientas seleccionadas para la comparativa han sido: *PhotoInfoEx*, *JHead*, *ExifTool*, *Exif Viewer* y *ExifPro Image Viewer*:

- ***PhotoInfoEx***: Es una aplicación de fotografía digital que permite editar o modificar ciertos metadatos de la información Exif o IPTC [ITPI07] de los archivos de imágenes en formato JPEG y TIFF.

  Las principales ventajas sobre la herramienta desarrollada son la mejor navegación sobre los archivos a examinar, la exportación de los metadatos obtenidos a Microsoft Excel y Microsoft Word y la impresión de los mismos.

  Como inconvenientes destacan:

  - Problemas en la extracción de metadatos Exif que no son acordes al 100 % con la especificación. Por ejemplo, se han detectado imágenes con datos en la etiqueta "DateTimeOriginal" erróneos para la fecha. *PhotoInfoEx*, en lugar de mostrar un error o la cadena tal como está almacenada, formatea los datos internamente y muestra otros distintos a los que posee la imagen, aparentemente correctos, cuando realmente no lo son.
  - No permite ningún tipo de análisis grupal de imágenes siendo éste un aspecto clave.
  - No muestra información referente a la imagen en miniatura almacenada en la información Exif por lo que no permite determinar si hay modificaciones con respecto a la imagen original.

- ***JHead***: Es una herramienta de línea de comandos muy potente que permite extraer y manipular la información Exif de los archivos JPEG [Wan10]. La única ventaja destacable de JHead frente a *Theia* es que permite la extracción de los metadatos IPTC y XMP, aunque éstos no sean utilizados por los dispositivos móviles.

  El principal inconveniente es que, al igual que *PhotoInfoEx*, no permite el análisis grupal de fotos, lo cual es fundamental. Asimismo, carece de interfaz gráfica, lo cual dificulta su uso, no posee funciones de geoposicionamiento y no realiza análisis de la imagen en miniatura.

- ***ExifTool***: Es una aplicación que permite la extracción y edición de metadatos en una gran variedad de formatos de archivos [Har05] tales como Exif, IPTC, XMP, JFIF. Además, permite decodificar información propia de los fabricantes (Maker Note) de gran cantidad de cámaras. Básicamente, las ventajas e inconvenientes de *ExifTool* con respecto a *Theia* son los mismos que con *JHead* y, consecuentemente, las conclusiones de la comparación.

- ***Exif Viewer***: Es un complemento para el navegador Firefox que permite extraer metadatos Exif, IPTC y XMP, de imágenes JPEG tanto locales como remotas [Ras07]. La principal ventaja de *Exif Viewer* con respecto a *Theia* es su facilidad y rapidez en la instalación, así como la facilidad de uso, teniendo en cuenta las grandes limitaciones que tiene. Otra de las ventajas es que permite el geoposicionamiento, además de en Google Maps y Google Earth, en Yahoo! Maps y en MSN Maps & Directions.

  El principal inconveniente, al igual que con todas las herramientas anteriormente comparadas, es que no permite un análisis de las imágenes en grupo. Asimismo, la forma de presentar la información y la interfaz son austeras y poco amigables. Por tanto, se concluye que es una herramienta con muchas menos posibilidades que *Theia*.

- ***ExifPro Image Viewer***: Es una aplicación que permite mostrar la información de un número muy limitado de etiquetas Exif de imágenes en formato JPEG [Kow10]. La principal ventaja de esta aplicación sobre todas las anteriormente tratadas (incluida *Theia*) es el navegador de archivos de las imágenes. Ofrece una inmensa cantidad de posibilidades para mostrar, agrupar y ordenar las imágenes del soporte de almacenamiento. Asimismo, es la más potente con respecto a la forma de mostrar las imágenes individuales.

  Con respecto a la extracción y tratamiento de los metadatos, sin duda, es la que más carencias tiene. Apenas extrae una veintena de etiquetas Exif, las cuales presenta de forma poco clara. No posee ningún tipo de funcionalidad de geoposicionamiento. Además, no permite el tratamiento grupal de los metadatos de las imágenes y no analiza la imagen en miniatura almacenada en la información Exif.

Posee una opción que posibilita la exportación de la información Exif incluida en fotografías JPEG a un fichero de texto. Esto facilita la posterior importación de la información del archivo de texto a otros formatos de bases de datos u hojas de cálculo y realizar consultas grupales sobre los datos exportados, pero la aplicación no permite directamente este tipo de operación, además de requerir al analista forense de conocimientos informáticos avanzados para realizar este tipo de tratamiento.

Por tanto, las conclusiones que se obtienen es que esta herramienta más que tratar metadatos Exif, es una herramienta para la visualización y clasificación de imágenes. El conjunto de metadatos Exif que obtiene es excesivamente limitado y, en ninguna circunstancia, es una aplicación válida para la tarea de análisis forense.

Una vez comparada *Theia* con otras con propósitos comunes, se puede concluir que, no habiendo ninguna que ofrezca todas las mejores posibilidades, *Theia* es la que ofrece una mayor funcionalidad, potencia y versatilidad a la tarea del analista forense. En la Tabla 6.1 se muestra una tabla comparativa de todas las herramientas evaluadas.

Tabla 6.1: Tabla comparativa entre las diferentes aplicaciones

Herramienta	**Plataforma**	**Interfaz**	**Versión Exif**	**Visualización de Datos Exif**	**Información Exif (Edición[a])**	**Formato de Metadatos**	**Software Libre**	**Software Gratuito**	**Análisis de Thumbnail**	**Observaciones**
Theia	- Windows - Mac OS - GNU/Linux	Gráfica	2.3	Organizada por IFD	No	Exif	Sí	Sí	Sí	- Interfaz amigable e intuitivo - Análisis grupal
PhotoInfoEx	- Windows	Gráfica	2.21	Organizada por IFD	Sí	Exif, IPTC	No	No	No	- Exportación de metadatos a otros formatos
Jhead	- Windows - Mac Os-X - GNU/Linux	Línea de comandos	No especificada	No organizada	Parcial	Exif, IPTC, XMP	Sí	Sí	No	- Difícil de usar - Extrae thumbnail pero no verifica cambios - Funcionalidades GPS reducidas
ExifTool	- Windows - Mac OS X - GNU/Linux	Línea de comandos	No especificada	No organizada	Sí	Exif, IPTC, XMP, JFIF	Sí	No	No	- Difícil de usar - Funcionalidades GPS reducidas
Exif Viewer	- Firefox Plugin	Gráfica	No especificada	Organizada por IFD	No	Exif, IPTC, XMP	Sí	Sí	No	- Metadatos de imágenes remotas
ExifPro Image Viewer	- Windows	Gráfica	No especificada	No organizada	No	Exif	No	No	No	- Información Exif limitada - Sin funcionalidad GPS

[a]Desde el punto de vista forense, es más robusto el evitar la modificación de imágenes accidentalmente.

Ninguna de las aplicaciones comparadas posee un tratamiento de imágenes en grupo, así como una extracción de metadatos Exif más completa y organizada. *Theia* no tiene como objetivo primordial la visualización de galerías de imágenes, sino favorecer y automatizar en la medida de lo posible la tarea del análisis forense para imágenes de dispositivos móviles con respecto a los metadatos. Este objetivo se consigue con mayor éxito en *Theia*.

6.3.1 Anomalías en Herramientas de Análisis Forense de Metadatos Exif

Después de analizar varios programas que tratan metadatos Exif se han observado diferentes criterios para la extracción de los datos ASCII en los ficheros JPEG. Analizando los distintos tipos de extracción de metadatos Exif se observa que existen distintas opciones:

1. En casos de violación de la especificación no mostrar los datos e indicar un error ya que no se sigue la misma. Esta opción es la más restrictiva, ya que no permite ningún tipo de ambigüedad.

Las siguientes opciones muestran alternativas que permiten la extracción de la información de la imagen a costa de pasar por alto el seguimiento estricto de la especificación Exif.

2. Extraer todos los datos del tipo ASCII hasta que se encuentre el primer nulo (0x00). Esta opción puede hacer que se generen errores graves, ya que si las cadenas ASCII no terminan en nulo, se pueden mostrar datos no pertenecientes a la etiqueta. Y en el peor de los casos, puede producir desbordamientos de memoria si en los bytes sucesivos a la etiqueta no existiera el valor nulo.

3. Extraer todos los datos teniendo únicamente en cuenta el tamaño indicado en la propia etiqueta. Esta es la opción menos restrictiva, ya que mostraría los caracteres ASCII del tamaño indicado, aunque éstos no cumplieran las restricciones de la especificación Exif.

4. Opción mixta entre 2 y 3. Es decir, extraer todos los datos teniendo el cuenta el tamaño de los mismos y separando las distintas cadenas teniendo en cuenta el nulo (0x00) como separador.

5. Extraer todos los datos de la etiqueta ignorando el tamaño indicado en el mismo. Es decir, si el tamaño es menor o igual a 4 bytes, extraer los cuatro bytes siguientes, y si es mayor de 4 bytes obtener el número de bytes indicados en el tamaño a partir del desplazamiento correspondiente. Para el tratamiento de los valores nulos (0x00) se debe elegir entre distintas opciones: (a) como espacios en blanco (lo cual puede generar problemas por hacerlo indistinguible con respecto al carácter ASCII de espacio en blanco), (b) ignorarlos (carácter vacío) o (c) sustituirlos por un carácter especial fuera del rango ASCII válido para Exif (0-127).

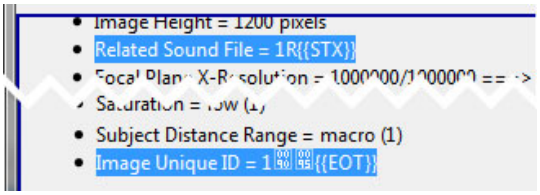
En *Theia* se optó por la opción 5, leer el número de datos especificado en la etiqueta y se reemplaza cada carácter nulo encontrado por una barra vertical ("|"). De esta manera se puede obtener toda la información posible de los datos de la etiqueta.

Se ha realizado un análisis de datos ASCII almacenados en distintas etiquetas Exif de imágenes de ejemplo, utilizando las siguientes herramientas enfocadas al análisis de metadatos Exif: *PhotoInfoEx* [Exs07], *Exif Viewer* [Ras07], *EXIFRead* [Lyo01], *ExifTool* [Har05] y *Jhead* [Wan10].

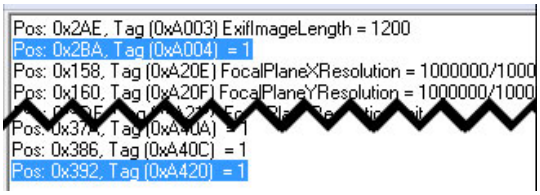
En la etiqueta "RelatedSoundFile" de una imagen se almacenan como datos erróneos "0x31005202" (ya que Exif indica que el tamaño de los datos de esta etiqueta debe ser 13). Aún teniendo esto en cuenta, *Exif Viewer* muestra "1R{{STX}}" como datos, lo cual indica que toma como opción la 5, ignorando el tamaño de los datos de la etiqueta (que en este caso es 1). También se observa que *Exif Viewer* ignora los caracteres nulos (0x00), ya que sólo muestra los 3 que no son nulos.

PhotoInfoEx, *EXIFRead*, *ExifTool* y *Jhead* muestran estrictamente los datos del número de elementos que dicta la etiqueta o hasta el primer nulo ("1"), es decir, utilizan la opción 2 o 3, no visualizando posible información sin inicializar o "basura" (ver Figura 6.7).

(a) *PhotoInfoEx*



(b) *Exif Viewer*



(c) *EXIFRead*

(d) *ExifTool*

(e) *Jhead*

Figura 6.7: Herramientas para el análisis de metadatos Exif

Independientemente de la forma de mostrar los datos Exif, hay un problema en la creación del archivo por parte del fabricante al no seguir fielmente la especificación. Por tanto, la opción tomada en la interpretación de estas anomalías es relevante y tiene consecuencias forenses.

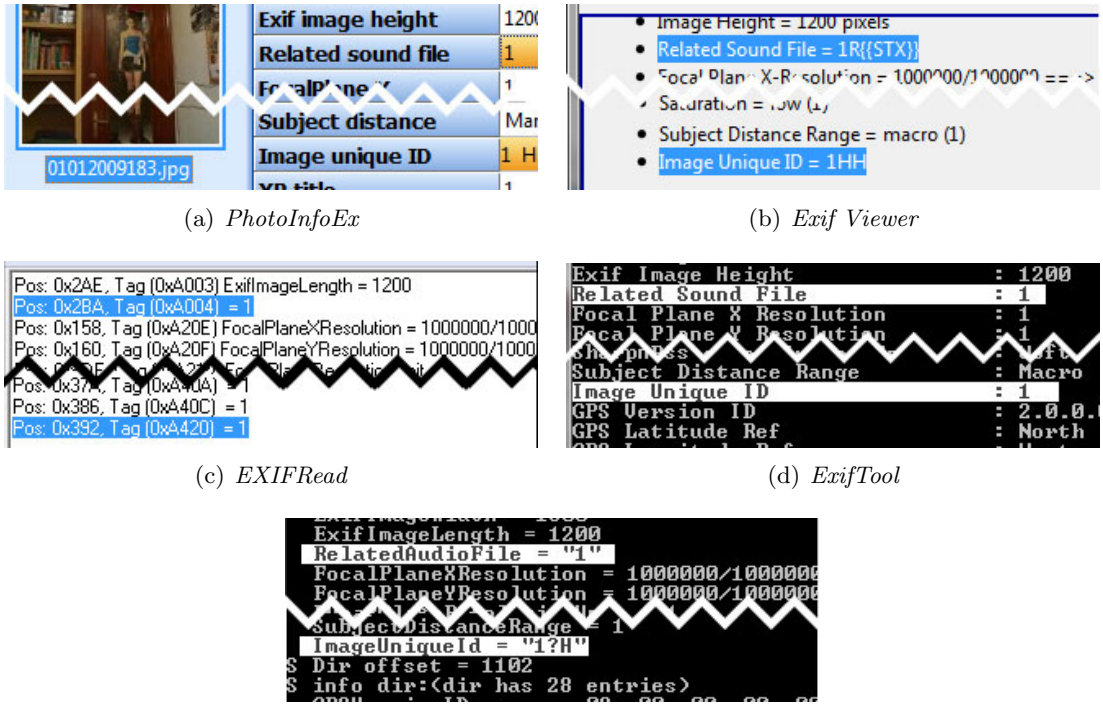
En el caso de la etiqueta *ImageUniqueID*, las herramientas *PhotoInfoEx*, *EXIFRead*, *Exif Tool* y *Jhead* muestran como valor de la etiqueta un "1", por lo que muestran los datos hasta el número de elementos que indica la etiqueta. En este caso dichas herramientas siguen también la opción 3. Por otra parte, *Exif Viewer* muestra "1□□{{EOT}}", es decir, cuatro caracteres, mostrando correctamente los valores ASCII que pertenecen al rango válido de la especificación Exif y un "cuadrado" con el valor en hexadecimal del byte en el interior cuando no pertenecen al rango válido (para este caso concreto los valores 90 y 95), como puede observarse en la Figura 6.7.

Para confirmar la opción tomada por 5 herramientas con respecto a los datos ASCII mostrados, se realizaron modificaciones al archivo con un editor hexadecimal. Se modificó la etiqueta "ImageUniqueID", poniéndole como tamaño de los datos 3 elementos (lo que sigue violando la especificación) y los datos a '0x31004848', por lo que la etiqueta completa es '0x20A4020003000000031004848'.

Una vez realizadas las modificaciones se revisaron nuevamente los datos de las etiquetas con las 5 herramientas (ver Figura 6.8), obteniendo los siguientes resultados:

- *EXIFRead* y *Exif Tool* muestran "1", siguiendo la opción 2.
- *PhotoInfoEx* y *Jhead* muestran "1 H" y "1?H", respectivamente, siguiendo la opción 3. Además, muestran los nulos (0x00) como el carácter espacio en blanco y el símbolo "?". Esto causa un grave problema a la hora de distinguir los caracteres ASCII espacio en blanco y "?" (0x20 y 0x3F) incluidos en el rango válido ASCII para Exif. En este caso concreto usando *PhotoInfoEx* y *Jhead* el analista forense no puede distinguir si los datos que almacena la etiqueta son "10x00H", "10x20H" o "10x3FH".
- *Exif Viewer* muestra los datos como "1HH". Con esto se puede ratificar que muestra los datos de la forma indicada en el caso 5, ignorando el tamaño de los datos de la etiqueta (que en este caso es 3). En este caso se confirma que *Exif Viewer* ignora los caracteres nulos (0x00).

Es conveniente volver a reseñar que el problema proviene del fabricante que no sigue la especificación Exif.



(a) *PhotoInfoEx* (b) *Exif Viewer*

(c) *EXIFRead* (d) *ExifTool*

(e) *Jhead*

Figura 6.8: Información una vez editada la etiqueta ImageUniqueID

Asimismo, se observa un error crítico en la herramienta *PhotoInfoEx* (que sigue la opción 3) al hacer clic en una etiqueta específica. Por ejemplo, en la etiqueta "ImageUniqueID" del ejemplo de la Figura 6.8, cuyo valor mostrado es "1 H" y su valor original es "1□□{{EOT}}". Simplemente el hecho de seleccionar el campo con el ratón y no hacer ninguna otra cosa más, provoca que el valor de la etiqueta pase a ser automáticamente "1" , tomando la herramienta ese campo como editado. Al cambiar de fotografía indica si se quieren guardar los cambios. Al aceptar la modificación de los metadatos de la fotografía, *PhotoInfoEx* no sólo modifica el valor de la etiqueta editada, sino que modifica todas las etiquetas de los metadatos Exif en los que se dan casos análogos al descrito, es decir, cambia igualmente otras etiquetas sin ser editadas por el analista. Esto puede ser perjudicial para la tarea del análisis forense, ya que modifica datos sin autorización, atentando así contra la integridad de la evidencia. En la Tabla 6.2 se muestra el *Message-Digest Algorithm 5* (MD5) calculado del fichero analizado antes y después de visualizarlo con *PhotoInfoEx*, el cual evidencia claramente cambios en el archivo.

Tabla 6.2: MD5 de la imagen antes y después de ser analizada con PhotoInfoEx

	Datos
Antes	4A07D9094BE9ADE0B719A2BF8AC1218C
Después	785B5670940811F0F58CE9D689FB2BFF

6.4 Síntesis del Capítulo

El objetivo de este capítulo ha sido la presentación de la herramienta *Theia*. Con el fin de automatizar y añadir funcionalidades extra que favorezcan el trabajo del analista forense se ha desarrollado *Theia*, una herramienta específica para el análisis forense de imágenes digitales. Inicialmente se ha realizado una presentación de las distintas funcionalidades de la herramienta. Seguidamente se ha mostrado la estructura y componentes básicos del diseño y la implementación de la herramienta.

Finalmente se ha realizado una comparativa de *Theia* con otras con fines similares. Sobre la valoración de *Theia* se han mostrado las diferencias con respecto a otras a la hora de la extracción de metadatos. Algunas otras aplicaciones permiten un rango mayor de formatos de metadatos. Sin embargo, la mayoría de las imágenes de dispositivos móviles están en formato JPEG/Exif, por lo que no hace que *Theia* tenga grandes desventajas con respecto a otras en ese aspecto. No se ha encontrado ninguna aplicación que mejore las funcionalidades específicas para el análisis forense que ofrece *Theia*, ya que ninguna permite realizar las siguientes operaciones:

- **Análisis grupal de imágenes**: Las herramientas estudiadas únicamente ofrecen información de metadatos de imágenes individuales. Cuando se quiere realizar análisis de grandes bancos de imágenes como los presentados en este capítulo esta funcionalidad es fundamental.
- **Identificación de manipulaciones de imágenes**: Mediante el uso de su imagen en miniatura se indica la probabilidad de manipulación de la imagen original.
- **Geoposicionamiento**: Tanto a nivel individual como grupal en Google Maps, además de geoposicionamiento individual en Google Earth.

Asimismo, se ha visto que algunas de las aplicaciones con las que se ha comparado *Theia* presentan anomalías a la hora de la extracción de los metadatos Exif. Este tema es de vital importancia ya que el analista forense necesita saber los metadatos insertos en las imágenes con exactitud y ser conocedor si ha habido algún problema en la extracción o en el seguimiento de la especificación Exif por parte del fabricante. Es inadmisible que una aplicación de extracción de metadatos haga cambios en los mismos sin permiso explícito del analista forense. Las anomalías, que pueden estar presentes en los metadatos Exif por causa de mal seguimiento de la especificación por parte de los fabricantes, deben ser informadas por medio de la aplicación al propio analista.

No se debe olvidar que todos los análisis realizados sobre metadatos Exif son desgraciadamente fácilmente vulnerables a modificaciones malintencionadas por terceros. Asimismo, la identificación de la fuente de la imagen con los metadatos depende totalmente de su inserción por parte del fabricante. Esto hace que la aplicación necesite de la colaboración de técnicas más robustas para la identificación de la fuente basadas en el contenido de la propia imagen y no en sus metadatos. No obstante, los metadatos aportan información útil para el analista forense, como por ejemplo la relacionada con el geoposicionamiento, la cual actualmente es imposible inferir mediante el contenido de la imagen y la evidencia de modificaciones posteriores a la captura de una fotografía (comparando la imagen con el thumbnail almacenado en su información Exif). Por tanto, se concluye que es necesario el estudio de nuevas técnicas para la identificación de fuente de una imagen, teniendo en cuenta que las basadas en los metadatos pueden ser de gran ayuda para otras técnicas.

Capítulo 7

Análisis Forense con Theia

Este capítulo presenta distintos análisis de bancos de imágenes realizados con *Theia*. Se comienza con el análisis de los metadatos Exif de un banco de imágenes, organizando el mismo según las distintas categorías de los metadatos. Posteriormente, se realiza un análisis de las anomalías Exif de otro banco de imágenes distinto al anterior, organizando el mismo según los distintos tipos de errores encontrados. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

7.1 Análisis de un Banco de Imágenes

A continuación se realiza el análisis de un conjunto de imágenes reales de dispositivos móviles utilizando las distintas funcionalidades de *Theia* descritas anteriormente. El objetivo del análisis es la búsqueda de datos de interés, patrones de valores o simplemente información estadística sobre los metadatos Exif del banco de imágenes. Las imágenes han sido obtenidas de dispositivos móviles de personas conocidas intentando buscar la máxima heterogeneidad posible con respecto a las marcas y los modelos de los dispositivos, así como contar con el mayor número de imágenes de cada uno de ellos. El banco de imágenes está formado por 3751 imágenes de 10 marcas y 91 modelos. En la Tabla 7.1 se muestran los modelos agrupados por marca con su correspondiente número de imágenes.

Tabla 7.1: Dispositivos móviles clasificados por marca y modelo

Marca	Modelo	Total
Apple	Ipad 2	19
	iPhone	24
	iPhone 3g	51
	iPhone 3GS	82
	iPhone 4g	49
HP	iPAQ hw6515	35
	IPAQ RX3000	35
HTC	8900	38
	Desire	41
	Desire HD	162
	Droid incredible	32
	Droid Incredible 2	27
	Evo 4g	38
	Hero	59
	MyTouch 4G	41
	TyTN ii	59
	Vodaphone HTCmagic	40
LG	CU720	31
	KF750	15
	KU990	30
	Ku990i	144
	Rumor	26
	VX-8550	13
	VX9700	30
Nokia	5230	19
	5300	100
	5530 XpressMusic	31
	5800	28
	6020	30
	6085	9
	6110 Navigator	35
	6120 Clasic	20
	6210 Navigator	29
	6230i	21
	6300	133
	6303 classic	35
	6600	36
	E61i	36
	E71	5
	N70	16
	N8	32
	N95	131
	N96	52
	N97	37
	N97 mini	54

Marca	Modelo	Total
Palm	Centro	28
	Pre	20
	Treo 680	22
Research In Motion	BB 8100	38
	BB 8300	31
	BB 8320	16
	BB 8330	34
	BB 8520	213
	BB 8900	37
	BB 9000	32
	BB 9550	30
	BB 9630	31
	BB 9800	30
Samsung	Caliber	8
	Captative	24
	Galaxy 3	33
	Galaxy S	15
	Galaxy S II	30
	H1	6
	Omnia 7	37
	Pixon	30
	SGH-F250L	4
	Start	39
	Wave	17
Sony Ericsson	C702	79
	C905	40
	K550i	13
	LT15i	11
	Satio	61
	T707	102
	Vivaz	16
	W580i	158
	W705	21
	W800i	39
	W910i	7
	X 10 Mini	10
	Z610i	61
Motorola	Atrix MB860	35
	Backflip mb300	47
	Cliq	30
	Defy mb525	22
	Droid	31
	Droid x	79
	Droid x2	54
	W377	20

A diferencia de los estudios realizados en otros trabajos relacionados, el número de modelos de cámaras utilizado es mucho mayor.

7.1.1 Análisis de la Información de Marca y Modelo

Ya que uno de los objetivos de *Theia* es la identificación de la fuente de la imagen, se utiliza "Query Set" para obtener el número de imágenes por marca y modelo. Al comparar estos datos con la Tabla 7.1 se valora el seguimiento que los fabricantes hacen sobre la inserción de estos dos metadatos. Los resultados de este análisis se muestran en la Tabla 7.2.

Tabla 7.2: Resultados del análisis de la información de marca y modelo

Marca	Modelo	Información Exif		Total
		Marca	Modelo	
Apple	Ipad 2	Apple	iPad 2	19
	iPhone	Apple	iPhone	24
	iPhone 3g	Apple	iPhone 3G	51
	iPhone 3GS	Apple	iPhone 3GS	82
	iPhone 4g	Apple	iPhone 4	45
	iPhone 4g	Apple	iPhone	4
HP	iPAQ hw6515		HP iPAQ hw6515	35
	iPAQ rx3000	HP	iPAQ rx3000	35
HTC	8900	HTC-8900	HTC-8900	38
	Desire	HTC	HTC Desire	41
	Desire hd	HTC	Desire HD	162
	Droid incredible	HTC	ADR6300	32
	Droid Incredible 2	HTC	ADR6350	27
	Evo 4g	HTC	PC36100	38
	Hero	HTC	HTC Hero	59
	MyTouch 4G	HTC	myTouch 4G	41
	TyTN ii	HTC	HTC_TyTN_II	59
	Vodaphone HTC magic	HTC	HTC Magic	38
	Vodaphone HTC magic	Vodafone	HTC Magic	2
LG	CU720	LG ELECTRONICS	CU720	31
	KF750	LG Electronics	KF750	15
	KU990	LG Electronics	KU990	30
	Ku990i	LG Electronics	KU990i	144
	Rumor	LG Electronics	RUMOR	26
	VX-8550	LG Electronics	VX-8550	13
	VX9700	LG Electronics	VX-9700	30
Motorola	Atrix mb860	Motorola	MB860	35
	Backflip mb300	Motorola	MB300	47
	Cliq	Motorola	MB200	30
	Defy mb525	Motorola	MB525	22
	Droid	Motorola	Droid	31
	Droid x	Motorola	DROIDX	79
	Droid x2	Motorola	DROID X2	54
	W377	Motorola	C261	20

Marca	Modelo	Información Exif		Total
		Marca	Modelo	
Nokia	5230	Nokia	5230	19
	5300	Nokia	5300	100
	5530 XpressMusic	Nokia	5530	31
	5800	Nokia	5800 Xpres	28
	6020			30
	6085	Nokia	0001	9
	6110 Navigator	Nokia	6110	35
	6120 Clasic	Nokia	6120c	20
	6210 Navigator	Nokia	6210 Navig	29
	6230i			21
	6300	Nokia	6300	133
	6303 classic	Nokia	6303 classic	35
	6600	Nokia	6600i-1c	36
	E61i	Nokia	E61i	36
	E71	Nokia	E71	5
	N70	Nokia	N70-1	16
	N8	Nokia	N8-00	32
	N95	Nokia	N95	66
	N95	Nokia	N95 8GB	65
	N96	Nokia	N96	52
	N97	Nokia	N97	37
	N97 mini	Nokia	N97 mini	54
Palm	Centro	Palm	Palm Centro	28
	Pre		Pre	20
	Treo 680		Treo 680	22
Research In Motion	BB 8100	RIM	BlackBerry 8100 Series	38
	BB 8300	Research In Motion	BlackBerry 8300	31
	BB 8320	Research In Motion	BlackBerry 8320	16
	BB 8330	Research In Motion	BlackBerry 8330	34
	BB 8520	Research In Motion	BlackBerry 8520	213
	BB 8900	Research In Motion	BlackBerry 8900	37
	BB 9000	Research In Motion	BlackBerry 9000	32
	BB 9550	Research In Motion	BlackBerry 9550	30
	BB 9630	Research In Motion	BlackBerry 9630	31
	BB 9800	Research In Motion	BlackBerry 9800	30
Samsung	Galaxy S II	samsung	GT-I9100	30
	Galaxy S	SAMSUNG	GT-I9000	15
	Caliber	SAMSUNG	SCH-R860	8
	Caliber	SAMSUNG	SPH-M570	8
	Captative	SAMSUNG	SGH-I897	24
	Galaxy 3	SAMSUNG	GT-I5800	33
	H1	SAMSUNG	Vodafone 360 Samsung H1	6
	Omnia 7	SAMSUNG	GT-I8700	37
	Pixon	SAMSUNG	GT-M8800	30
	SGH-F250L	SAMSUNG	SGH-F250L	4
	Start	Samsung	GT-S5230	39
	Wave	SAMSUNG	GT-S5333	17

Marca	Modelo	Información Exif		Total
		Marca	Modelo	
Sony Ericsson	C702	Sony Ericsson	C702	79
	C905	Sony Ericsson	C905	40
	K550i	Sony Ericsson	K550i	13
	LT15i	Sony Ericsson	LT15i	11
	Satio	Sony Ericsson	U1a	26
	Satio	Sony Ericsson	U1i	35
	T707	Sony Ericsson	T707	102
	Vivaz	Sony Ericsson	U5i	16
	W580i	Sony Ericsson	W580i	158
	W705	SONY ERICCSON	W705	21
	w800i	Sony Ericsson	W800i	39
	w910i	Sony Ericsson	W910i	7
	X 10 Mini	SEMC	X10a	10
	Z610i	Sony Ericsson	Z610i	61

Como puede observarse en la Tabla 7.2, existen distintos ejemplos en los cuales no existe coherencia entre los modelos de móviles y la información de marca y modelo en los metadatos Exif. Por ejemplo, algunas imágenes del "IPhone 4" tienen como valor de modelo "IPhone", con lo cual las hace indistinguibles con respecto a las realizadas con el "IPhone" (modelo diferente a IPhone 4). De forma similar, se puede observar que el Nokia 6085 almacena como modelo el valor "Nokia 0001". Y en el Nokia N95 se almacenan 66 imágenes con el valor "N95" y 65 con el valor "N95 8GB", dado que son dos versiones del mismo modelo. En el caso del fabricante Palm se observa que los modelos "Treo 680" y "Centro" tienen NULL en la etiqueta "Make".

La primera conclusión de este análisis es positiva, ya que se puede apreciar un alto grado de seguimiento por parte de los fabricantes a la hora de almacenar los valores para marca y modelo (un 98,64 %), aunque existen algunos casos en los que la información almacenada no sigue ningún tipo de patrón coherente.

Otro aspecto destacado de este análisis es la escasa uniformidad de los propios fabricantes a la hora de añadir la información de marca y modelo en las etiquetas Exif, ya que no utilizan siempre la misma cadena para la marca o incluso puede haber una misma cadena para distintos modelos (caso IPhone y IPhone4), lo cual puede dar pie a graves errores en la identificación. Por ejemplo, Sony Ericsson utiliza dependiendo del modelo la cadena "EMC", "Sony Ericcson" o "SONY ERICCSON" para almacenar la marca, o "Research In Motion" utiliza indistintamente en los modelos de "BlackBerry" la cadena "RIM" o "Research In Motion" para el mismo fin. Adicionalmente, se ha detectado que móviles que han sido diseñados específicamente para una compañía de móviles, como es el caso particular del HTC Magic que tiene un modelo para Vodafone, en algunos casos

almacenan en la etiqueta "Make" el nombre de la compañía de móviles. Esta información puede ser relevante desde el punto de vista forense ya que revela datos sensibles del usuario como es en este caso la compañía a la que pertenece el móvil con el que fue tomada la imagen.

Asimismo, se ha utilizado "Advanced Query" para identificar cuáles son las imágenes que no contienen la información de marca y modelo. En el resultado se obtienen las imágenes concretas y son todas las de los modelos "Nokia 6020" y "Nokia 6230".

7.1.2 Información de las Etiquetas Image y Exif

Se analizan las etiquetas Exif que se encuentran en los bloques "Image Info" y "Exif Info" con "Query Set", examinando cuáles son las imágenes que no poseen información en ninguno de estos dos bloques. El resultado de este análisis es que todas las imágenes poseen información en los dos campos.

Además en el bloque "Image Info" se analiza con "Query Set" el campo "Software Used". Este campo puede ser de importancia para el análisis forense, ya que puede aportar datos como el software de creación de la imagen. Los resultados obtenidos revelan uniformidad en el nombrado de versiones por parte de cada fabricante. Entre los distintos fabricantes la discordancia es total. También se destaca que el software utilizado parece variar en función de la operadora del móvil para un mismo modelo. Por ejemplo, para un Sony Ericcson W580i la etiqueta "Software Used" tiene valores del estilo:

- "R8BE001 prgCXC1123474_ORANGE_LA 0.0": para la operadora Orange.
- "R8BE001 prgCXC1123362_GENERIC_L 0.0": para cualquier otra operadora.

7.1.3 Análisis de la Información GPS

Se analizan las etiquetas Exif que se encuentran en el bloque "GPS Info", examinando cuáles son las imágenes que no poseen información en este bloque. Este análisis es muy subjetivo, ya que depende de si el terminal tiene GPS integrado, de que el usuario lo tenga activado y de que permita la inserción de la información GPS en el momento de la toma. Aún así, los resultados aportan que 2918 de las imágenes analizadas no poseen información GPS y 823 sí. Dentro de este bloque de información se analizan las imágenes con información GPS pero que no poseen las etiquetas de latitud y longitud detectándose 191 imágenes. Con "Advanced Query" se detectan 144 del móvil LG KU990i, 16 del Sony Ericsson Vivaz, 30 del Motorola Droid X y que una pertenece al modelo Sony Ericsson Satio.

Con respecto al modelo LG KU990i en sus especificaciones técnicas se indica que no posee sistema GPS. Aún así, almacena en todas las imágenes la etiqueta "GPSVersionID" con valor "Version 2.3", el cual no es obligatorio si no hay información GPS. El Sony Ericsson Vivaz y el Sony Ericcson Satio sí poseen GPS (concretamente un *Assisted Global Positioning System* (A-GPS)). El Sony Ericsson Vivaz permite geoetiquetar las imágenes pero no almacena la información GPS correspondiente. Finalmente, el Sony Ericcson Satio almacena las etiquetas "GPSVersionID" con valor "0.0", "GPSAltitudeRef" con valor "Sea level" y "GPSAltitude" con valor "0". Este último caso puede ser debido a que la fotografía se tomó con el sistema A-GPS desactivado o que el usuario no permitió la inserción de información GPS en la imagen. Aún así, carece de sentido rellenar las tres etiquetas anteriores con valores aparentemente erróneos.

Las 632 imágenes que tienen datos de geoposicionamiento se pueden observar en la Figura 7.1.

Figura 7.1: Visualización de imágenes en Google Maps

7.1.4 Análisis de la Información de Imagen en Miniatura

Se analizan las etiquetas Exif que se encuentran en el bloque "Thumbnail Info" con "Query Set", examinando cuáles son las imágenes que no poseen información en este bloque. El resultado de este análisis, que se muestra en la Tabla 7.3, revela que 2883 imágenes (el 76,85 %) posee información de la imagen en miniatura, frente a 868 que no la posee; de estas últimas, 456 son del fabricante Research In Motion.

Tabla 7.3: Análisis de la imagen en miniatura

Marca	Modelo	Thumbnail		Total
		N	S	
Apple	Ipad 2	13	6	19
	Iphone	5	19	24
	Iphone 3g	2	49	51
	iphone 3GS	19	63	82
	iphone 4g		49	49
Research In Motion	BB 8100	38		38
	BB 8300	31		31
	BB 8330	34		34
	BB 8520	213		213
	BB 8900	8	29	37
	BB 9000	29	3	32
	BB 9550	30		30
	BB 9630	31		31
	BB 9800	30		30
HP	iPAQ hw6515	35		35
	iPAQ rx3000	35		35
HTC	8900	2	36	38
	desire	13	28	41
	desire hd	4	158	162
	droid incredible	14	18	32
	Droid Incredible 2	20	7	27
	evo 4g	11	27	38
	Hero		59	59
	myTouch 4G	5	36	41
	tyTN ii	28	31	59
	vodaphone htc magic	6	34	40
LG	CU720		31	31
	KF750	1	14	15
	KU990		30	30
	ku990i		144	144
	rumor	5	21	26
	VX-8550		13	13
	VX9700		30	30
Motorola	Atrix mb860	1	34	35
	backflip mb300	1	46	47
	Cliq	1	29	30
	Defy mb525	2	20	22
	droid		31	31
	droid x		79	79
	Droid x2		54	54
	W377	20		20
Palm	Centro		28	28
	Pre	4	16	20
	Treo 680		22	22

Marca	Modelo	Thumbnail		Total
		N	S	
Nokia	5230		19	19
	5300	12	88	100
	5530 XpressMusic		31	31
	5800		28	28
	6020	30		30
	6085	9		9
	6110 Navigator		35	35
	6120 Clasic		20	20
	6210 Navigator	2	27	29
	6230i	6	15	21
	6300	7	126	133
	6303 classic		35	35
	6600		36	36
	E61i		36	36
	e71		5	5
	N70		16	16
	N8		32	32
	N95	65	66	131
	N96	2	50	52
	N97	1	36	37
	N97 mini		54	54
Samsung	Galaxy S II		30	30
	Galaxy S		15	15
	Caliber	3	5	8
	Captative		24	24
	Galaxy 3	1	32	33
	Omnia 7		37	37
	Pixon		30	30
	SGH-F250L	4		4
	start		39	39
	Wave	2	15	17
Sony Ericsson	C702	11	68	79
	c905		40	40
	k550i	1	12	13
	LT15i		11	11
	Satio	1	60	61
	T707		102	102
	vivaz		16	16
	W580i		158	158
	W705		21	21
	w800i	4	35	39
	w910i	4	3	7
	X 10 Mini		10	10
	Z610i		61	61

Con este análisis se encuentra que el fabricante Research In Motion no almacena la imagen en miniatura en la información Exif en modelos inferiores al BB 8900. Asimismo, algunos modelos puntuales de otros fabricantes como Palm Centro, Treo 680, HP iPAQ hw6515, Motorola W377, Nokia 6020 y Nokia 6085 no poseen imagen en miniatura. El uso de la imagen en miniatura es recomendado en Exif, por lo que no tenerlo, aunque no viola la especificación, es una situación bastante inusual.

Cabe destacar otro análisis para identificar el modo de compresión de la imagen en miniatura, ya que aunque la imagen sea JPEG comprimida, la imagen en miniatura puede no estarlo. El resultado de este análisis muestra que la imagen en miniatura de todas las imágenes está en formato comprimido JPEG.

Por último, se hace un análisis de identificación de modificaciones realizadas a las imágenes comparando la imagen en miniatura almacenada en la información Exif con el thumbnail generado a partir de la imagen. Este análisis se basa en el cálculo del RMS del resultado de la comparación de las dos imágenes en miniatura. Los valores RMS procedentes de esta comparación se clasifican en 3 grupos: "no modificadas" para aquellas imágenes que tengan un valor RMS inferior a 5, "posiblemente modificadas" para aquellas imágenes con valor RMS entre 5 y 25, y "modificadas" para aquellas imágenes con valor RMS superior a 25. El resultado de este análisis se puede observar en la Tabla 7.4.

Tabla 7.4: Resultados del análisis de imágenes modificadas.

Clasificación	**Observaciones**	**Subtotal**	**Total**
Sin imagen en miniatura		872	872
No modificadas	Imagen rotada	5	322
	Diferente tamaño	131	
	Mismo tamaño y orientación	186	
Posiblemente modificadas	Imagen rotada	26	2335
	Diferente tamaño	428	
	Mismo tamaño y orientación	1881	
Modificadas	Imagen rotada	2	222
	Diferente tamaño	100	
	Mismo tamaño y orientación	120	
Total		3751	3751

En las Figuras 7.2, 7.3 y 7.4 se pueden observar ejemplos de imágenes de cada grupo obtenidas en este análisis.

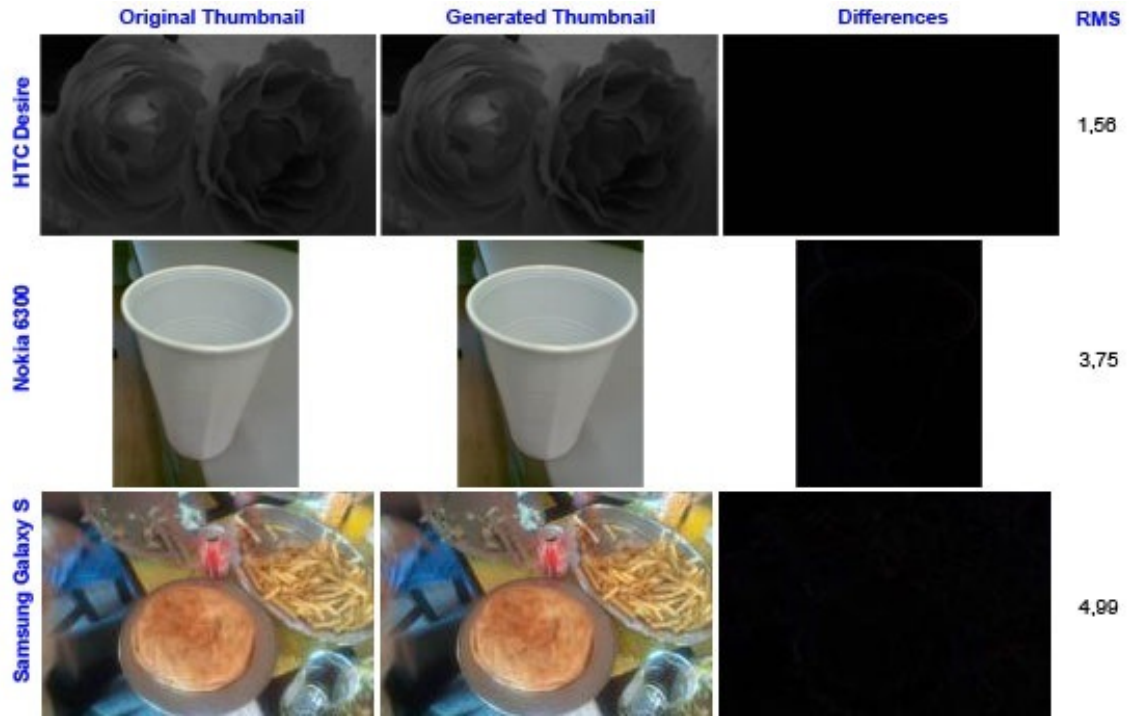


Figura 7.2: Imágenes no modificadas (RMS < 5)

Figura 7.3: Imágenes posiblemente modificadas (RMS entre 5 y 25)

Figura 7.4: Imágenes modificadas (RMS > 25)

7.1.5 Análisis de la Información Maker Note

En este apartado se analizan las etiquetas Exif que se encuadran en el bloque "Maker Note Info", examinando con "Query Set" las imágenes que no poseen información en este bloque. El resultado de este análisis muestra que el 0 % de las imágenes poseen información "Maker Note Info". Esto revela que en los dispositivos móviles los fabricantes no insertan ningún tipo de información propia, aunque esta afirmación requiere de un estudio a fondo antes de poder realizar una extrapolación a todo el conjunto de dispositivos móviles.

7.1.6 Análisis de la Información de Interoperabilidad

Al analizar las etiquetas Exif que se encuentra en el bloque "Interoperability Info" con 'Query Set" se encuentra que 2082 imágenes (un 50,5 %) contienen este tipo de información, frente a 1669 que no.

7.2 Anomalías en el Seguimiento de la Especificación Exif

Tras el análisis binario de varias imágenes se han detectado casos en los que no se sigue la especificación al 100 %, aún indicando en su cabecera lo contrario. A continuación, se muestran ejemplos en los que el fabricante asegura que su imagen sigue la especificación Exif y realmente no es así. Por tanto, se estima necesario que las herramientas de extracción de metadatos contemplen esta situación, intentando reconocer la mayoría de los errores más comunes detectados tras el análisis binario.

7.2.1 Ejemplos de Anomalías

En una fotografía tomada con un Samsung Galaxy S se detecta que una entrada del directorio IFD0 es 0x1001020008000000C6000000, cuyo significado se muestra en la Tabla 7.5.

Tabla 7.5: Etiquetas 0th IFD con anomalías

Dispositivo Móvil	Primera entrada 0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento
Samsung Galaxy S	0x1001020008000000C6000000	Model (0x0110)	ASCII (0x0002)	8 (0x00000008)	198 bytes (0x000000C6)
Nokia N70	0x04A002000100000031005202	Related Audio File (0x04A0)	ASCII (0x0002)	1 (0x00000001)	0x31005202
	0x20A402000100000031909504	Unique Image ID (0xA420)	ASCII (0x0002)	1 (0x00000001)	0x31909504

Como se observa del desplazamiento 0xC6, el valor de la etiqueta "Model" es "GT-I9000" y tiene longitud 8 como se indica en la cabecera. A simple vista todo es correcto, pero siendo estrictos, esta imagen no cumple al 100 % la especificación Exif 2.2, ya que se indica que el tipo es 2 (ASCII terminado en NULL - 0x00) y esta cadena no termina en NULL. Para almacenar "GT-I9000" se necesitan 9 elementos (8 caracteres ASCII + 1 NULL) y no 8 como indica la entrada del directorio.

Otro caso se da en una imagen de un teléfono móvil Nokia N70, que asegura seguir la especificación Exif 2.2. Las etiquetas que se analizan son las siguientes: 0xA004 ("Related Audio File") y 0xA420 ("Unique Image ID").

La entrada de la etiqueta "Related Audio File" es 0x04A0020001000000031005202, cuya interpretación se muestra en la Tabla 7.5.

Según la especificación Exif la etiqueta "Related Audio File" es de tipo ASCII y posee 13 elementos, es decir, 0x0000000D, pero como se puede observar en la Tabla 7.5, realmente almacena 0x00000001, es decir, 1 elemento, lo cual viola claramente la especificación por dos razones. Primero, porque la etiqueta "Related Audio File" indica que el tamaño de los datos tiene que ser 13 bytes y segundo, porque la especificación Exif indica que el tamaño mínimo de los datos tiene que ser 4 bytes.

Una vez detectado que este archivo no sigue la especificación, los datos que se almacenan son 0x31005202. Este valor es un 1 en ASCII, seguido del valor nulo 0x00, R en ASCII y el valor 0x02 (STX en ASCII). Este hecho puede generar problemas para los programas que extraen la información Exif por la incoherencia entre la especificación y los datos almacenados. Los visores de información Exif deberían tomar un criterio uniforme para la extracción de cadenas ASCII.

Independientemente de la forma de mostrar los datos de los visores Exif, hay un problema en la creación del archivo por parte del fabricante al no seguir fielmente la especificación. Por tanto, cualquier opción tomada podría tener consecuencias forenses ya que este tipo de anomalías pueden ser utilizadas con fines maliciosos por herramientas anti-forenses.

Otro caso de anomalía se da en el mismo teléfono móvil (Nokia N70) y en la etiqueta "Unique Image ID". Según la especificación Exif la etiqueta "Unique Image ID" es de tipo ASCII y posee 33 elementos, es decir, 0x00000021, pero como se puede observar en la Tabla 7.5, realmente almacena 0x00000001, es decir, 1 elemento, lo cual viola claramente la especificación. Una vez visto que este archivo no sigue la especificación, el valor de la etiqueta almacenada es 0x31909504, teniendo en cuenta 4 bytes, ya que el análisis revela que el quinto byte es el comienzo de otra etiqueta. Este hecho hace que se viole de nuevo la especificación, ya que en el tipo ASCII es obligatorio que termine en nulo (0x00) y en este caso el nulo no aparece en la cadena. Asimismo, existe otra violación de la especificación

ya que los caracteres ASCII son de 7 bits (rango de 0-127, 0x00-0x7F), por lo que los caracteres 0x90 y 0x95 están fuera de lo que permite la especificación.

En este apartado se han mostrado algunos ejemplos de anomalías detectadas tras el análisis binario manual de las imágenes, pero no han sido las únicas ya que se han encontrado más en otras imágenes y otras etiquetas como "Exif version", "Meetering Mode", "Exposure Program", "DateTimeOriginal", ...

Por tanto, se puede concluir que muchos de los fabricantes no siguen fielmente las especificaciones Exif, indicando en el propio archivo lo contrario, lo cual puede producir graves problemas en la extracción de los metadatos de las imágenes por medio de aplicaciones, así como problemas de interoperabilidad entre distintos dispositivos.

7.2.2 Clasificación de los Errores en las Etiquetas Exif

Theia es capaz de reconocer 10 tipos de errores bastante comunes en las etiquetas Exif, numerándolos del 0 al 9. La Tabla 7.6 muestra los 10 tipos de errores detectados.

Tabla 7.6: Tipos de errores detectados por Theia

Tipo	Descripción
0	Carácter de Terminación con Datos Tipo ASCII
1	Error en el Tipo de Datos
2	Error en el Número de Datos
3	Imagen en Miniatura no Encontrada
4	Etiquetas Duplicadas
5	Error en el Formato de la Fecha
6	Etiqueta Fija no Encontrada
7	Etiqueta no Permitida
8	Datos Exif no Encontrados
9	Error en Datos GPS

Estos errores son devueltos en una estructura de la siguiente forma:

{"**Id de tipo**" { "**Nombre de Etiqueta**": "**Datos del error encontrado**"}}

donde:

- **Id de tipo:** Cadena de caracteres que toma los siguientes valores: Image, Thumbnail, Exif, Interoperability o GPS. Se añade IFD1 a la cadena del error cuando la imagen en miniatura no se encuentra o se encuentra en la etiqueta *Maker Notes*.

- **Nombre de Etiqueta**: Cadena de caracteres del nombre de la etiqueta correspondiente dentro de la clase. Tiene dos casos especiales para los dos errores que no son de etiqueta:
  - Cuando no hay información Exif, tiene el valor "EXIF".
  - Si es un error en la búsqueda de la imagen en miniatura (si no está o está en la etiqueta *Maker Notes*), tiene el valor "Thumbnail".
- **Datos del error encontrado**: Tupla de 3 elementos. Dependiendo del tipo de error, se usan todos o parte de los campos. Los 3 campos de la tupla son:

  (*id*, *datos error 1*, *datos error 2*)

  - *id*: Entero con valor de 0 a 9 que indica el tipo de error encontrado según la tabla de la siguiente sección.
  - *datos error 1*: Campo que puede usarse o no, dependiendo del tipo de error. Por defecto, si no se usa, tiene el valor "None".
  - *datos error 2*: Al igual que el anterior sólo se usa por cierto tipos de errores (en este caso, sólo errores de tipo 1 y 2). Por defecto, si no se usa, el valor es "None".

Según la especificación Exif 2.3, si el tamaño total de los datos de una etiqueta es inferior o igual a 4 bytes, éstos se incluyen en el campo denominado desplazamiento; en caso contrario, este campo apunta a la posición de los mismos, tratándose como un desplazamiento a partir del inicio de la cabecera TIFF.

A continuación se realiza una descripción de los distintos tipos de errores Exif detectados por *Theia*:

- **Error Tipo 0**: Según la especificación Exif 2.3, las etiquetas cuyos elementos son tipo ASCII deben acabar en carácter nulo (0x00). Sin embargo, si la etiqueta es tipo "Undefined" y contiene un dato tipo ASCII, no hay restricción alguna sobre la terminación.
- **Error Tipo 1**: Según la especificación Exif 2.3, ciertas etiquetas tienen asociado un tipo de dato predeterminado. Si no se usa este tipo de dato, no se sigue la especificación.

  Los tipos soportados son:
  - 1 = BYTE ⇒ Entero sin signo de 8-bits.
  - 2 = ASCII ⇒ 1 BYTE que contiene 7-bits de código ASCII.

  - 3 = SHORT ⇒ Entero sin signo de 16-bits.
  - 4 = LONG ⇒ Entero sin signo de 32-bits.
  - 5 = RATIONAL ⇒ 2 LONG. El primero es el numerador y el segundo el denominador.
  - 7 = UNDEFINED ⇒ 1 BYTE que puede tomar cualquier valor dependiendo de la definición del campo.
  - 9 = SLONG a 32-bits ⇒ Entero con signo de 32-bit.
  - 10 = SRATIONAL ⇒ 2 SLONG. El primero es el numerador y el segundo el denominador.

  Existen casos en los que puede haber dos tipos de datos: "Short" y "Long", que *Theia* representa como tipo "11".

- **Error Tipo 2**: Al igual que en el tipo de datos, existen etiquetas que sólo pueden tener un determinado número de elementos (a veces se especifican varios números de elementos posibles). Si esto no se cumple en la propia etiqueta, la especificación es incorrecta. En los demás casos puede haber cualquier número de elementos.

- **Error Tipo 3**: La imagen en miniatura se almacena en el IFD 1 del fichero JPEG. El tener esta imagen resumen supone un gran avance de cara a posibles análisis forenses, ya que la imagen original puede haber sido modificada y podemos sacar ciertas conclusiones de la imagen en miniatura. El estándar Exif 2.3 no obliga a que cada imagen tenga su propia imagen en miniatura guardada, pero lo recomienda. Existen modelos de cámaras que almacenan la imagen en miniatura en la etiqueta *Maker Notes*. Sin embargo, en los diferentes análisis no se ha encontrado ninguna fotografía de móvil que tenga esta característica. *Theia* detecta este tipo de error especificando en cada caso si no se ha encontrado la imagen en miniatura o si ésta está almacenada en la etiqueta *Maker Notes*.

- **Error Tipo 4**: Este error se presenta cuando una etiqueta aparece dos o más veces en el fichero JPEG. Suele presentarse en imágenes que han sido manipuladas por herramientas de edición y retoque de imágenes que no actualizan correctamente los metadatos.

- **Error Tipo 5**: El formato de la fecha en la especificación Exif 2.3 tiene dos variantes:

  - Con fecha y hora: "YYYY:MM:DD HH:MM:SS"
  - Sólo con fecha: "YYYY:MM:DD"

  En ambos casos, si no se conoce el dato se deben dejar los caracteres de datos en blanco y los separadores en su correspondiente lugar. Si no se siguen estos patrones para la fecha o la hora, se viola la especificación Exif 2.3.

- **Error Tipo 6**: Según el tipo de compresión JPEG utilizada, la especificación Exif obliga la inclusión de determinadas etiquetas en los metadatos. Se consideran 4 tipos de compresión JPEG que difieren en diferentes características, siendo sus etiquetas obligatorias:

  - CHUNKY:

    TagChImage = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, StripByteCounts, XResolution, YResolution, ResolutionUnit, ExifOffset, ExifVersion, FlashPixVersion, ColorSpace]

    TagChThumbnail = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, Strip ByteCounts, XResolution, YResolution, ResolutionUnit]

  - PLANNAR:

    TagPlImage = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, StripByteCounts, XResolution, YResolution, ResolutionUnit, ExifOffset, ExifVersion, FlashPixVersion, ColorSpace]

    TagPlThumbnail = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, Strip ByteCounts, XResolution, YResolution, PlanarConfiguration, ResolutionUnit]

  - YCC:

    TagYCCImage = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, Strip ByteCounts, XResolution, YResolution, ResolutionUnit, YCbCrSubSampling, YCbCr Positioning, ExifOffset, ExifVersion, FlashPixVersion, ColorSpace]

    TagYCCThumbnail = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, StripByteCounts, XResolution, YResolution, ResolutionUnit, YCbCrSubSampling]

  - COMPRESSED:

    TagCompressedImage = [XResolution, YResolution, ResolutionUnit, YCbCrPositioning, ExifOffset, ExifVersion, ComponentsConfiguration, FlashPixVersion, ColorSpace, PixelXDimension, PixelYDimension]

    TagCompressedThumbnail = [Compression, XResolution, YResolution, ResolutionUnit, JPEGInterchangeFormat, JPEGInterchangeFormatLength]

  La forma de conocer el tipo de compresión JPEG realizada se basa en la consulta de determinadas etiquetas que dan datos, como el 0x106 (*PhotometricIntepretation*).

Esta etiqueta ayuda a distinguir entre COMPRESSED que no lo contiene y los demás. Asimismo, el propio valor distingue entre YCC (YCbCr) y CHUNKY o PLANNAR (RGB). Para diferenciar entre CHUNKY y PLANNAR se usa la etiqueta 0x11C (*"PlannarConfiguration"*).

- **Error Tipo 7**: Existen determinadas etiquetas Exif cuya inclusión en los metadatos no está permitida para los distintos tipos de compresión JPEG. A continuación, se muestran los 4 tipos de compresión de JPEG con sus correspondientes etiquetas no permitidas.

  - CHUNKY:

    notTagChImage = [JPEGInterchangeFormat, JPEGInterchangeFormatLength, YCbCr-Coefficients, YCbCrSubSampling, YCbCrPositioning, ComponentsConfiguration, CompressedBitsPerPixel, PixelXDimension, PixelYDimension, InteroperabilityOffset]

    notTagChInteroperability = [InteroperabilityIndex]

    notTagChThumbnail = [JPEGInterchangeFormat, YCbCrCoefficients, JPEGInterchangeFormatLength, YCbCrSubSampling, YCbCrPositioning]

  - PLANNAR:

    notTagPlImage = [JPEGInterchangeFormat, JPEGInterchangeFormatLength, YCbCr- Coefficients, YCbCrSubSampling, YCbCrPositioning, ComponentsConfiguration, CompressedBitsPerPixel, PixelXDimension, PixelYDimension, InteroperabilityOffset]

    notTagPlInteroperability = [InteroperabilityIndex]

    notTagPlThumbnail = [JPEGInterchangeFormat, YCbCrCoefficients, JPEGInterchangeFormatLength, YCbCrSubSampling, YCbCrPositioning]

  - YCC:

    notTagYCCImage = [ComponentsConfiguration, CompressedBitsPerPixel, JPEGInterchangeFormat, JPEGInterchangeFormatLength, PixelXDimension, PixelYDimension, InteroperabilityOffset]

    notTagYCCInteroperability = [InteroperabilityIndex]

    notTagYCCThumbnail = [JPEGInterchangeFormat, JPEGInterchangeFormat Length]

  - COMPRESSED:

    notTagCompressedImage = [ImageWidth, ImageLength, BitsPerSample, Compression, PhotometricInterpretation, StripOffsets, SamplesPerPixel, RowsPerStrip, StripByteCounts, PlanarConfiguration, JPEGInter changeFormat, YCbCrSubSampling, JPEGInterchangeFormatLength]

notTagCompressedThumbnail = [ImageWidth, ImageLength, BitsPerSample, PhotometricInterpretation, SamplesPerPixel, StripByteCounts, RowsPerStrip, StripOffsets, PlanarConfiguration, YCbCrSubSampling]

- **Error Tipo 8**: Este tipo de error se presenta en imágenes que han sido procesadas por herramientas de edición y retoque de imágenes o al borrar los metadatos manualmente. Si no se detectan metadatos en formato Exif (estos datos puede que no estén o que estén en otro formato) o no se reconoce el JPEG mediante su cabecera (en hexadecimal 0xFFD8), no se puede realizar el análisis binario y se tiene que analizar mediante otros métodos más sofisticados para poder sacar la máxima información posible.

- **Error Tipo 9**: Se han detectado que algunos modelos, algunos muy conocidos como los distintos modelos Iphone de Apple, tienen errores relacionados con las etiquetas pertenecientes al GPS. Según el estándar Exif 2.3, siempre que aparezca la etiqueta *GPSInfo*, que indica que hay datos GPS y su desplazamiento, es obligatorio incluir en el IFD perteneciente al GPS la etiqueta *GPSVersionID*. Si esta etiqueta no aparece se incumple el estándar. Adicionalmente, etiquetas como *GPSLatitude* y *GPSLongitude* que en conjunto indican la posición geográfica donde ha sido tomada la fotografía pueden no incluirse y no incumplirse el estándar. También se presenta el caso en el que estas etiquetas se encuentran pero tiene sus valores (rational) a 0 (0/1 exactamente), lo que indica que el dato es muy probablemente erróneo. Debido a todo esto se ha estructurado la salida del error de la siguiente manera:

  - 0: No se encuentra la etiqueta *GPSVersionID*.
  - 1: No se encuentra la etiqueta *GPSLatitude* o *GPSLongitude*.
  - 2: Las etiquetas *GPSLatitude* o *GPSLongitude* tienen un valor de 0.

7.2.3 Análisis Individual de Imágenes por Tipo de Error

Una vez realizada la clasificación de los errores más comunes que se pueden presentar por incumplimiento de la especificación Exif, se muestra un ejemplo de cada tipo de error analizando a nivel binario imágenes reales tomadas con dispositivos móviles. Este análisis tiene como objetivo comprobar el seguimiento de la especificación por parte de los fabricantes.

Cabe destacar que en el almacenamiento binario de la información Exif se pueden seguir dos tipos de alineaciones: "Intel" (*little endian*) o "Motorola" (*big endian*). Este aspecto está especificado en el propio archivo y hay que tenerlo en cuenta para la interpretación binaria de la información de los metadatos.

- **Análisis de Errores Tipo 0**: Utilizamos para el análisis una fotografía tomada con un Samsung Galaxy S. La Tabla 7.7 muestra la etiqueta "Model", donde se observa que los datos son tipo ASCII y por tanto debe acabar en nulo. Realizando un desplazamiento hasta la posición de los datos se lee en ASCII "GT-I9000", observándose que no acaba en carácter nulo (0x00), violando la especificación Exif 2.3.

Tabla 7.7: Análisis del error tipo 0: Etiqueta Model

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x1001020008000000C6000000	Model (0x0110)	ASCII (0x0002)	8 (0x00000008)	198 bytes (0x000000C6)	"GT-I9000"

- **Análisis de Errores Tipo 1**: En este caso se utiliza para el análisis una fotografía realizada con un Nokia N97. La Tabla 7.8 muestra la etiqueta *Gain Control* con datos de tipo "Rational". Por tanto, el contenido del desplazamiento no puede ser de otro tipo. En la Tabla 7.8 se observa que el número de elementos de la etiqueta es 1 y que los datos están ubicados en el campo desplazamiento al ser inferior o igual a 4 con tipo de datos "Short". Esta etiqueta no cumple la especificación al no contener un dato tipo "Rational" como lo señala Exif 2.3, obteniéndose un error en el tipo de datos.

Tabla 7.8: Análisis del error tipo 1: Etiqueta Gain Control

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x07A4030001000000002000000	Gain Control (0xA407)	Short (0x0003)	1 (0x00000001)	-	"2"

- **Análisis de Errores Tipo 2**: En este caso se utiliza para el análisis una fotografía realizada con un HTC Desire HD. La Tabla 7.9 muestra la etiqueta *GPSVersionID* de las etiquetas pertenecientes al GPS con un número de datos fijado a 4. Si la etiqueta no tiene especificado este número de elementos se habrá violado la especificación Exif 2.3. Se observa que no coincide el número de elementos con los especificados para esa etiqueta. Por tanto, no se cumple la especificación Exif 2.3.

Tabla 7.9: Análisis del error tipo 2: Etiqueta GPSVersionID

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x00000001000000030202000	GPSVersionID (0x0000)	Byte (0x0001)	3 (0x00000003)	-	[2][2][0]

Un caso excepcional se presenta cuando el número de elementos especificado en la etiqueta multiplicado por el tamaño del dato es superior a 4 bytes y en la etiqueta se observa que calculando este tamaño total es igual o inferior a 4 bytes. Según la especificación Exif 2.3, si el número de bytes que ocupa el dato de una etiqueta es inferior o igual a 4 bytes, el campo desplazamiento de la etiqueta debe contener el dato; de lo contrario, este campo contiene el desplazamiento al dato. *Theia* en este caso devuelve en el campo valor de la etiqueta una lista con los dos posibles valores del dato: el valor del desplazamiento y el valor que se corresponde con el número de elementos de la especificación, tomando el contenido como desplazamiento (en vez de tomarlo como dato).

Para este análisis se utiliza una imagen tomada con un Nokia N70. La etiqueta *RelatedSoundFile*, mostrada en la Tabla 7.10, debe tener obligatoriamente 13 elementos como valor del número de datos. Si la etiqueta no tiene especificado este número de elementos se viola la especificación Exif 2.3. Se observa que el número de elementos es 1. Por tanto, al ser inferior o igual a 4 bytes los datos están en el campo desplazamiento. Teniendo en cuenta lo anterior, el valor del dato es 1. Sin embargo, este valor es incorrecto ya que esta etiqueta indica que hay un archivo de audio relacionado con la imagen (8 bytes del nombre + '.' + 3 bytes para la extensión). Si se ignora el número de elementos que dice la etiqueta y se usa el de la especificación Exif 2.3 (13 elementos), el campo desplazamiento apunta a la posición del dato, observándose que el desplazamiento es muy superior al tamaño del fichero (0x02520031 $>$ 0x00057FB0), por lo que se concluye que tanto el dato como el desplazamiento han sido grabados con un dato erróneo. *Theia* muestra los 2 resultados si el desplazamiento está dentro del rango del tamaño del fichero. En este caso, al ser mayor que el propio fichero, el campo desplazamiento no muestra más que el valor del dato.

Tabla 7.10: Análisis del error tipo 2: Etiqueta RelatedSoundFile

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x04A0020001000000310052O2	RelatedSoundFile (0xA004)	ASCII (0x0002)	1 (0x00000001)	0x02520031	1

- **Análisis de Errores Tipo 3**: Para este análisis se usan 100 imágenes de un móvil BlackBerry Bold 8900. Ninguna de ellas posee imagen en miniatura. Los móviles BlackBerry no almacenan la imagen en miniatura en modelos anteriores al 9000. Se ha intentado encontrar alguna razón para esto por parte del fabricante, pero lo único que se ha conseguido es usuarios frustrados por no incluirlo y ninguna respuesta por parte del equipo técnico de Research in Motion.

- **Análisis de Errores Tipo 4**: Para este análisis se utiliza una imagen tomada con un Iphone 4G. La Tabla 7.11 muestra la etiqueta *Orientation* y la Figura 7.5 el fichero editado en hexadecimal con la herramienta "HxD". En la Figura se observa que la etiqueta *Orientation* tiene 2 apariciones consecutivas y que ambas etiquetas son idénticas. Asimismo, puede observarse que este archivo utiliza para el almacenamiento de los datos Exif la alineación Motorola ("MM").

Tabla 7.11: Análisis del error tipo 4: Etiqueta Orientation

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x011200030000000100010000	Orientation (0x0112)	Short (0x0003)	1 (0x00000001)	-	1

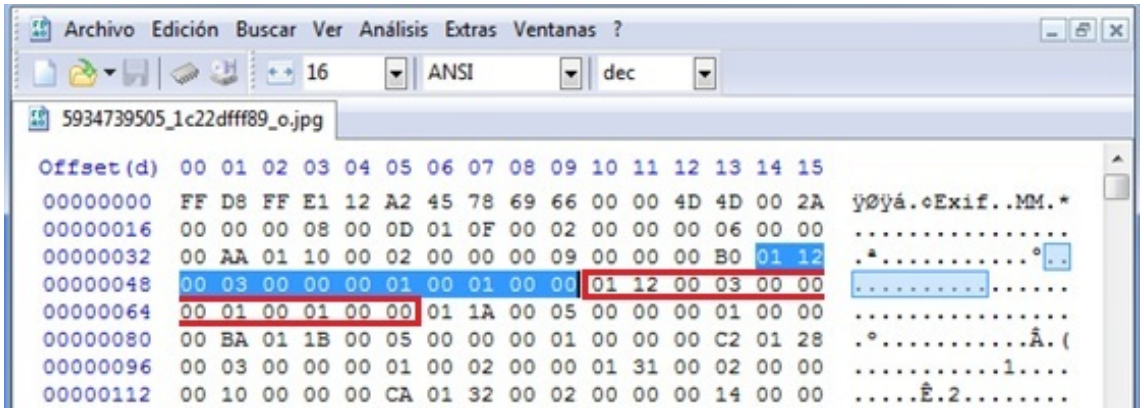


Figura 7.5: Análisis del error tipo 4: Etiqueta Orientation

- **Análisis de Errores Tipo 5**: La Tabla 7.12 presenta la etiqueta *DateTimeDigitized* de una imagen bajada de internet tomada con un Samsung SGH-F250L. En los datos de la etiqueta (2010:1010:10 12:53:12) se observa claramente que no tiene el formato de fecha y hora, además de ser completamente erróneo y, por tanto, no cumple la especificación. También se detecta otro error (tipo 0: carácter de terminación con datos tipo ASCII) ya que al ser una cadena ASCII no termina en "0x00".

Tabla 7.12: Análisis del error tipo 5: Etiqueta DateTimeDigitized

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x900400020000001400000 11A0	DateTimeDigitized (0x9004)	ASCII (0x0002)	20 (0x00000014)	0x000011A0	2010:1010:10 12:53:12

Un caso especial se presenta en una imagen tomada con un LG KF750 en la etiqueta *DateTimeOriginal* mostrada en la Tabla 7.13. Se observa que el dato de fecha y hora es correcto pero no está en el formato del estándar Exif 2.3.

Tabla 7.13: Análisis del error tipo 5: Etiqueta DateTimeOriginal

0th IFD	Etiqueta	Tipo	Número de Elementos	Desplazamiento	Valor
0x900300020000001400001232	DateTimeOriginal (0x9004)	ASCII (0x0002)	20 (0x00000014)	0x000011A0	8 Oct 2010 11:28:20

- **Análisis de Errores Tipo 6**: En este caso se analizan todas las etiquetas de una imagen tomada con un Samsung H1, observándose que no se encuentra la etiqueta *XResolution* (0x11A), que es obligatoria para cualquier tipo de almacenamiento de la imagen. Al faltar esta etiqueta no se cumple la especificación, independientemente del tipo de almacenamiento de la imagen.

- **Análisis de Errores Tipo 7**: Al analizar la misma imagen tomada con un Samsung H1 se detecta la existencia en los metadatos de la etiqueta *ImageWidth* (0x100), que no está permitida para el tipo de almacenamiento "COMPRESSED". Esta imagen utiliza este tipo de almacenamiento, violando la especificación Exif.

- **Análisis de Errores Tipo 8**: Un caso de datos Exif no encontrados es el de imágenes tomadas con un Ipad 2. En este JPEG no se detecta información Exif, pero sí en otro tipo de formato de metadatos: el *International Color Consortium* (ICC).

- **Análisis de Errores Tipo 9**: En este caso se analiza una imagen tomada con un Iphone 4G de Apple. Como en todas las imágenes encontradas de Iphone con información GPS, no se incluye la etiqueta *GPSVersionID*. Esto es un incumplimiento del estándar Exif 2.3, aunque los datos de posición sean correctos.

  Al analizar una fotografía tomada con un Sony Ericsson Satio se observa que aparece la etiqueta *GPSVersionID*, pero no las etiquetas *GPSLatitude* ni *GPSLongitude*. Esto no es un incumplimiento de la especificación, pero la información GPS que se tiene es insuficiente para geolocalizar la fotografía. Comentar, por último, que los datos de

otras etiquetas de información GPS en esta imagen tienen alta probabilidad de ser erróneos, ya que muchos de sus valores parecen estar rellenos con valores sin sentido (valores rellenos con 0).

7.2.4 Experimentos

Los experimentos se realizan utilizando una base de datos de 4000 imágenes de diferentes modelos de móviles de las marcas más conocidas. Esta base de datos se divide en 2 grandes grupos: el primero de ellos constituido por las imágenes "recolectadas", que son las que se han obtenido directamente de los dispositivos móviles y puede considerarse que no han sido manipuladas posteriormente a la captura de la imagen. El segundo grupo pertenece a las imágenes "descargadas de internet", que son imágenes menos fiables, ya que existen algunas en las que incluso se ha eliminado la información Exif que contenían. Primero se realiza un análisis por marcas, seguidamente un análisis de los terminales más problemáticos, posteriormente un análisis de los errores más comunes y finalmente un análisis de las etiquetas más propensas a tener fallos. La Tabla 7.14 presenta los fabricantes de los dispositivos móviles de los que se han obtenido las imágenes.

Tabla 7.14: Número de imágenes utilizadas por fabricante

Marca	Total
Apple	400
BlackBerry	600
HP	100
HTC	500
LG	400
Motorola	300
Nokia	600
Palm	100
Samsung	400
Sony Ericsson	600
Total	4000

La Figura 7.6 muestra para cada fabricante el porcentaje de imágenes con anomalías en los metadatos Exif. Como se observa en la Figura 7.6, marcas como BlackBerry (99,79 %), HP (100 %) y Palm (100 %) tienen un alto grado de error en el seguimiento de la especificación Exif 2.3. Samsung y LG con un 87,97 % y 89,62 %, respectivamente. Apple tiene un gran porcentaje de imágenes erróneas debido a errores tipo 9, ya que sus imágenes con información GPS no siguen el estándar. Sony Ericsson, en cambio, es la que mejor sale parada con un 4,69 % de imágenes erróneas, que es un porcentaje muchísimo menor que la siguiente marca en tener menos errores, que es Nokia con un 54,4 % de imágenes erróneas.

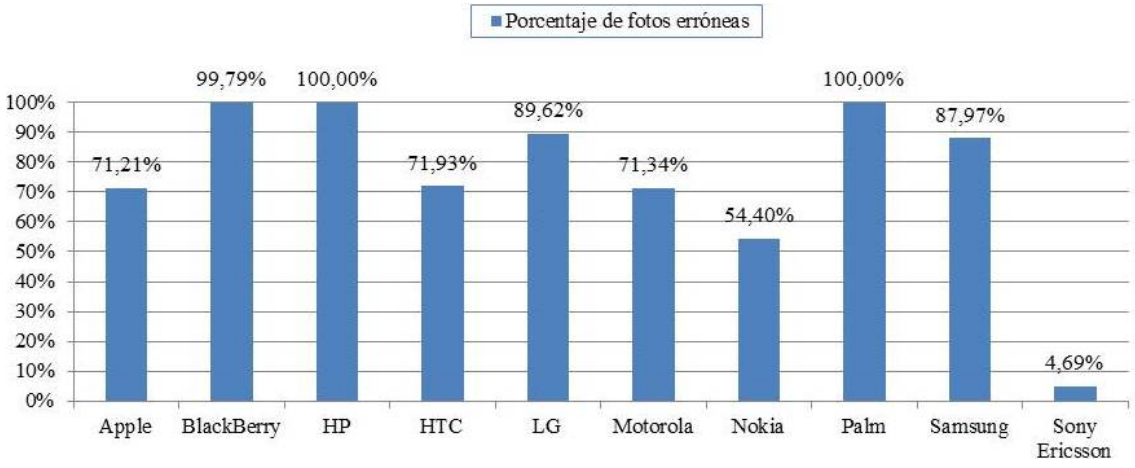


Figura 7.6: Porcentaje de imágenes erróneas por marca

Debido a la posible manipulación de las imágenes descargadas de Internet, la Figura 7.7 presenta un análisis de imágenes recolectadas que será mucho más preciso, aunque el número de imágenes se reduzca considerablemente. En este caso, las marcas que peor salen paradas son BlackBerry, LG, Motorola y Samsung, ya que todas las imágenes recolectadas de estas marcas incumplen la especificación Exif. Al igual que en el caso anterior, Sony Ericsson es el fabricante que menor tasa de imágenes erróneas tiene, con un 0,23 %. Apple tiene un 65,56 % de imágenes que no cumplen el estándar y HTC tiene un 63,31 %. Finalmente, se destaca que Nokia es el fabricante más beneficiado con respecto al análisis anterior, ya que mejora el seguimiento de la especificación Exif en un 14,75 %.

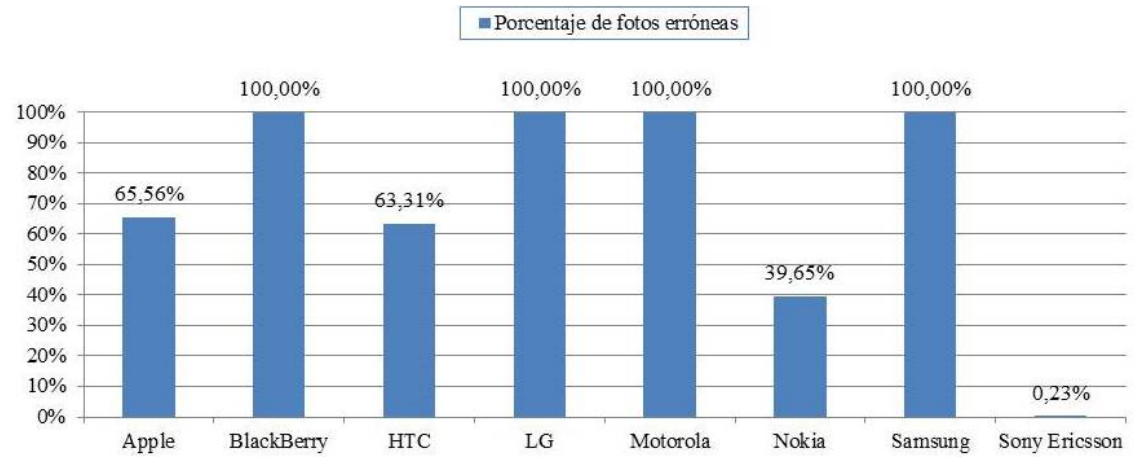


Figura 7.7: Porcentaje de imágenes recolectadas erróneas por marca

En el siguiente experimento se analizan modelos concretos usando todas las imágenes de la base de datos (descargadas de internet y recolectadas). En los resultados se han encontrado muchos modelos en los que todas sus imágenes contienen algún error, lo que indica que tales modelos no almacenan los metadatos siguiendo la especificación Exif al 100 %. Algunos ejemplos concretos de modelos conocidos son: BlackBerry 9630, Motorola Droid/Droid X, Nokia N70/N95/N97, Nokia N8, Samsung Galaxy S2. La Figura 7.8 presenta un análisis más exhaustivo centrándose en el número de errores por modelo. Como el número de errores puede variar de una imagen a otra, se utiliza para la estadística el número medio de errores en el total de las imágenes. Estos porcentajes han sido sacados de los 15 terminales con más errores.

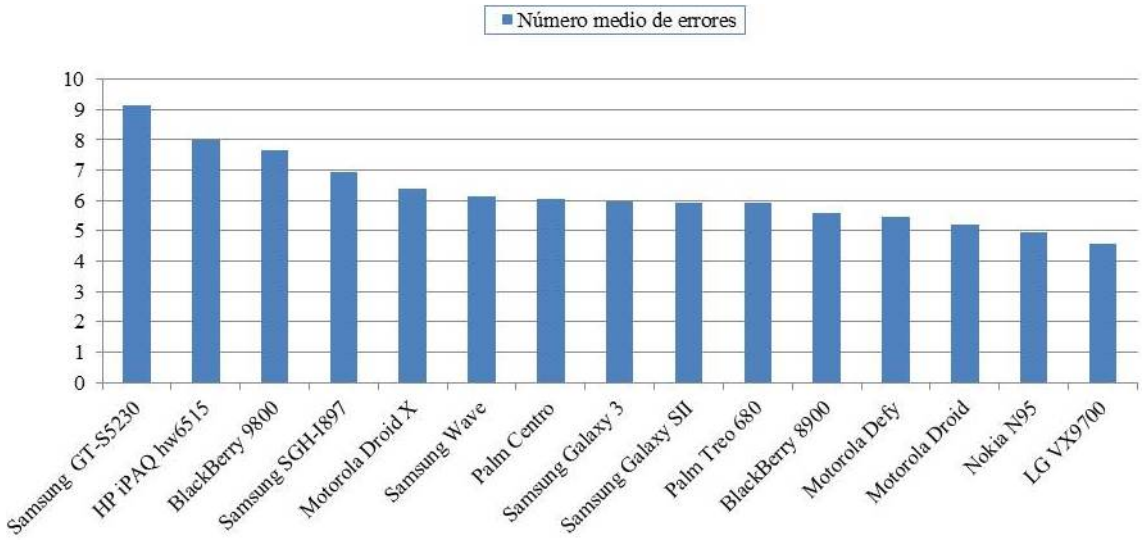


Figura 7.8: Número medio de errores por modelo

Como ya se ha comentado, existen diferentes tipos de errores reconocidos, desde unos que no permiten la extracción de ningún tipo de información (falta de información Exif) hasta otros que aportan información errónea, duplicada, inexistente o que no siguen con total fidelidad el formato especificado en Exif. En todo caso debe quedar claro que supone un incumplimiento de la especificación, salvo los casos de imagen en miniatura no encontrada (error tipo 3) y error en los datos GPS (error tipo 9) para los casos en los que se devuelve 1 o 2 en los datos del error (falta de *GPSLatitude* y *GPSLongitude* o valor 0 de los mismos). La Figura 7.9 presenta los porcentajes de cada error de todas las imágenes (descargadas y recolectadas), observándose que el más común es el de tipo 6, es decir, etiqueta fija no encontrada, que puede ocurrir varias veces en una misma imagen. Otros errores comunes son los de tipo 1 (error en el tipo de datos), que también puede ocurrir varias veces en una imagen y el de tipo 3, imagen en miniatura no encontrada, que no es un incumplimiento del estándar, pero sí una recomendación (este error sólo puede ocurrir una vez por imagen).

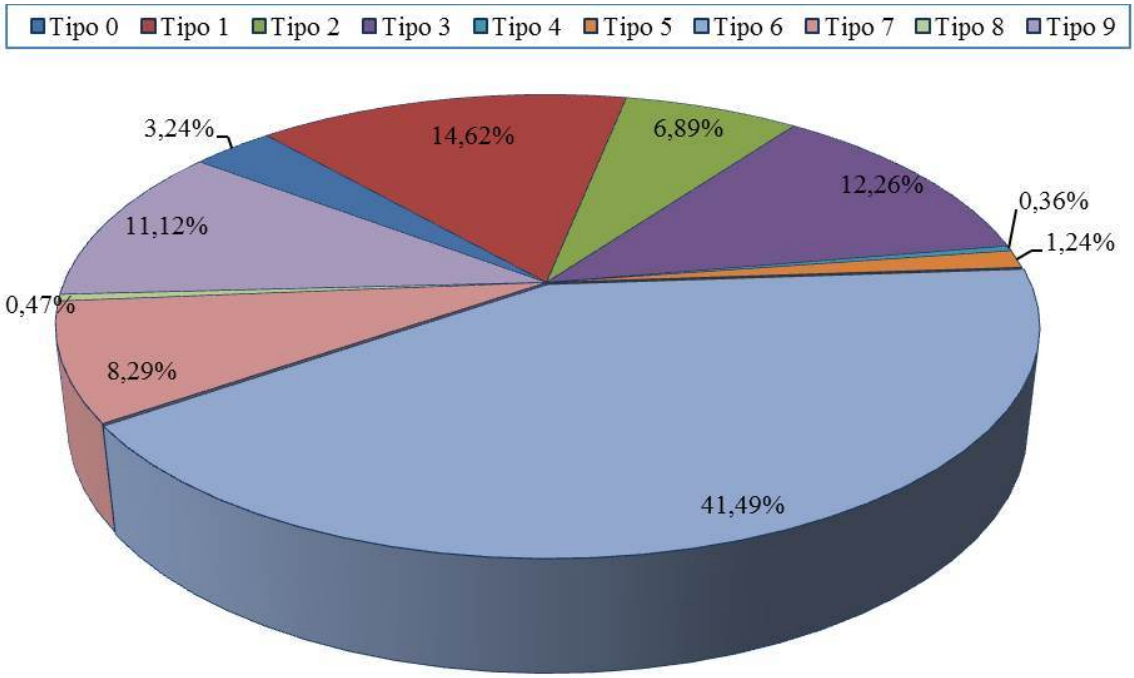


Figura 7.9: Porcentaje de imágenes con errores en metadatos por tipo de error

Al analizar sólo las imágenes recolectadas se observa que los porcentajes varían y algunos tipos de errores ya no están presentes (ver Figura 7.10). Los siguientes tipos de errores han desaparecido por completo: 4 (etiqueta duplicada), 5 (error en el formato de la fecha) y 8 (datos Exif no encontrados). Al tomar sólo las imágenes recolectadas el número de errores de tipo 6 ha disminuido considerablemente. Estos cambios pueden deberse claramente a que las imágenes descargadas de internet tienen más probabilidad de haber tenido modificaciones por aplicaciones externas. Los errores de tipo 3 (imagen en miniatura no encontrada) y de tipo 9 (error en datos GPS) se mantienen proporcionales al porcentaje de imágenes reducido.

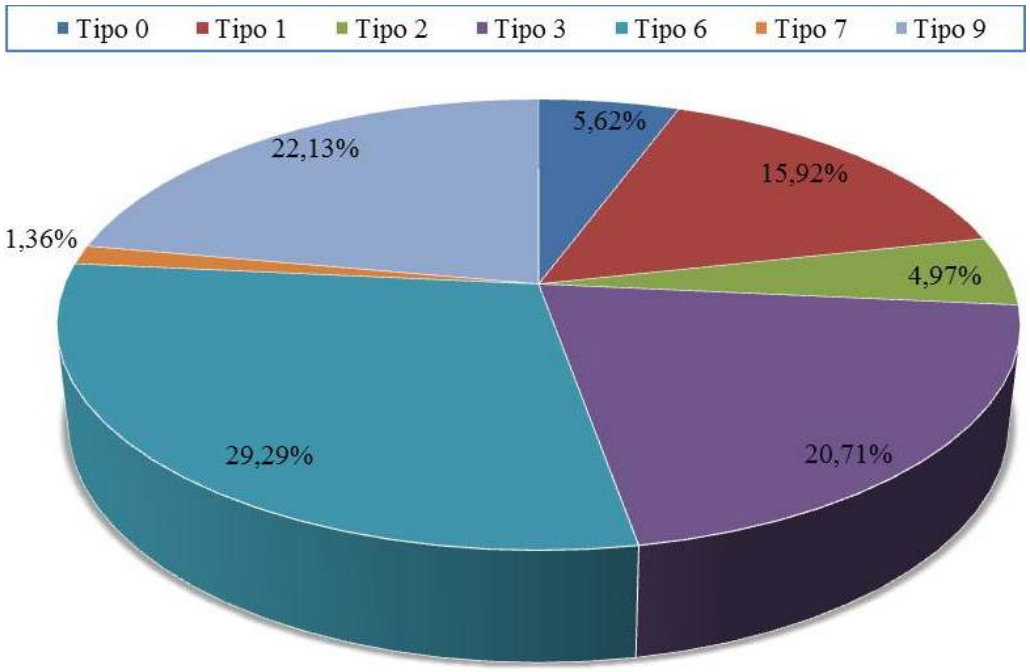


Figura 7.10: Porcentaje de imágenes recolectadas erróneas por tipo de error

En la Figura 7.11 se realiza un análisis por marca y tipo de error, contabilizando el número de errores totales de la marca como el 100 %. Hay que tener en cuenta que un mismo error puede aparecer varias veces en una imagen. Por tanto, éste se contabiliza más de una vez. Para este análisis se utiliza la base de datos de imágenes completa. El error dominante es el tipo 6. Este error suele aparecer muchas veces por imagen. También son altos los porcentajes de errores de tipo 1, 3, 7 y 9. Este último con un gran porcentaje en determinadas marcas que no siguen el estándar en los datos GPS como son Apple, LG y Motorola. Los errores como el 3 o el 8 sólo aparecen una vez por imagen.

Debido a la característica de estos errores y para equilibrar los porcentajes, la Figura 7.12 presenta un análisis en el cual sólo se contabiliza por cada imagen un error de cada tipo.

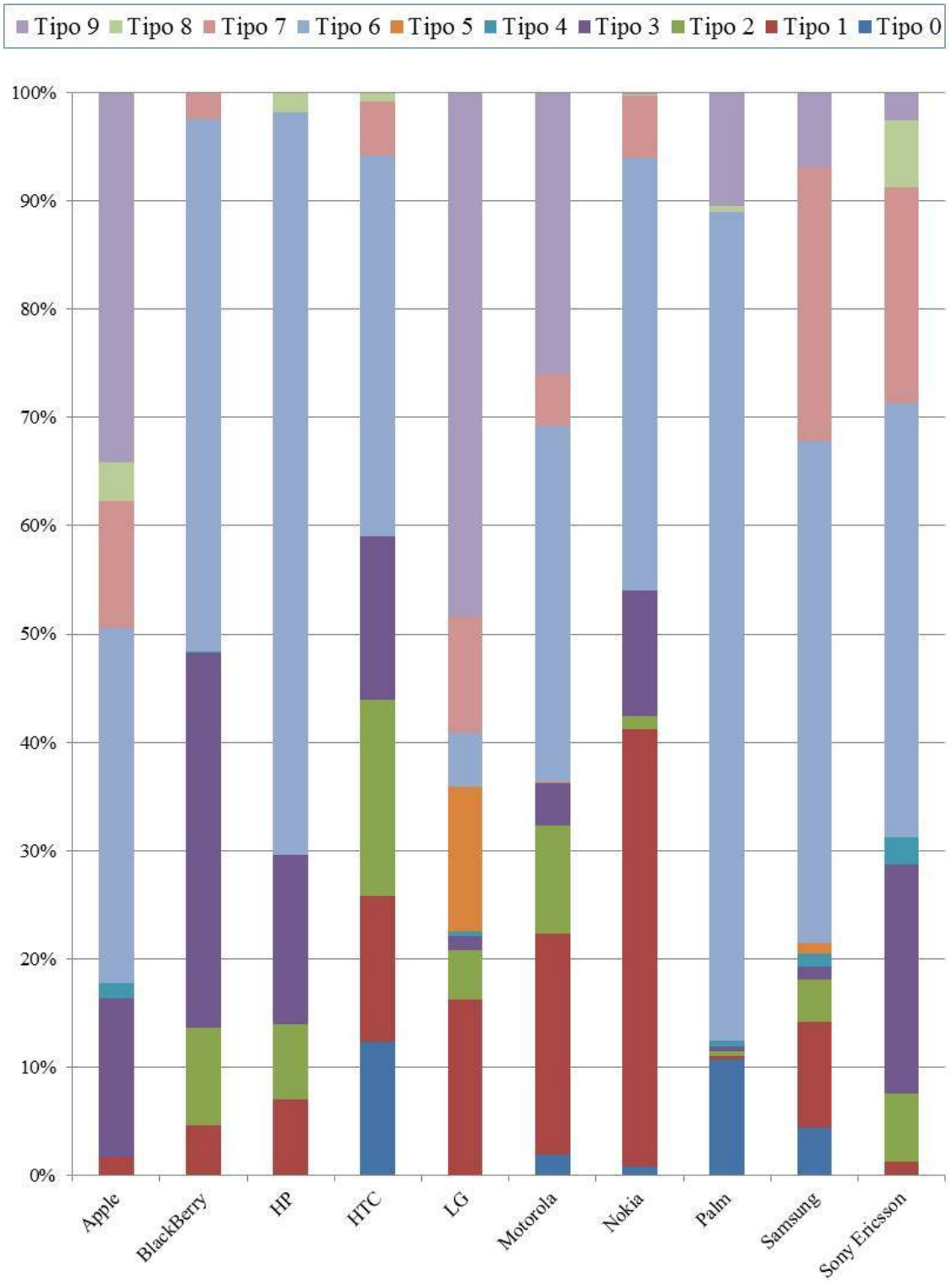


Figura 7.11: Porcentaje de imágenes erróneas por marca y tipo de error

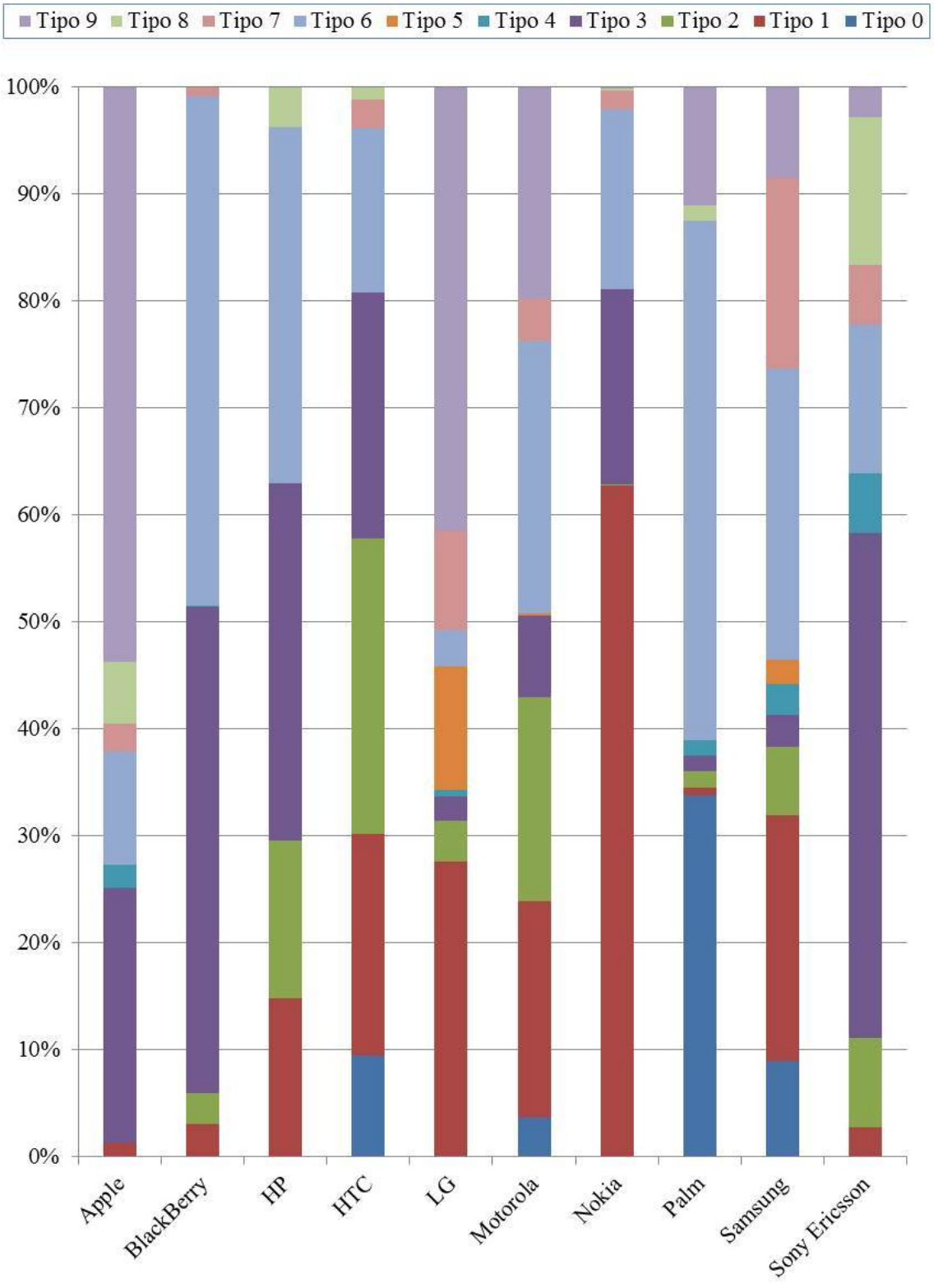


Figura 7.12: Porcentaje de imágenes erróneas por marca con un error por tipo

Los errores de tipo 6 y 7 son los más afectados en este análisis, demostrándose que estos errores son los que suelen aparecer más veces repetidos en una imagen. Aún así se observa que el tipo 6 sigue siendo muy común en la mayoría de las marcas. Destacar que el error de tipo 3, que sólo aparece una vez por imagen, se hace mucho más notorio, ya que este análisis no reduce su número de apariciones, siendo en algunas marcas como Apple, BlackBerry, HP o Sony Ericsson del orden del 25 al 50 % de los errores encontrados. También se hace mucho más importante el error 9 en ciertas marcas como Apple y LG, superando en Apple el 50 % de los casos. Este error sólo puede aparecer cuando el móvil tiene la opción de geolocalización activada, por lo que seguramente si tuvieran todas las imágenes con geolocalización activada se tendría un porcentaje del orden del 100 % en algunas marcas. En los terminales Nokia es muy común (del orden del 62 %) el error de tipo 1. Si se analiza el número de apariciones sólo se reduce en 3 unidades respecto al experimento anterior, lo que implica que hay una determinada etiqueta común en los móviles Nokia, donde está siendo mal utilizado el tipo datos de la etiqueta.

La Figura 7.13 presenta un análisis de las etiquetas más propensas a tener errores. En ella se observa que la etiqueta más propensa a tener errores es *FlashPixVersion* (en hexadecimal "0xA000" con 772 errores en el total de las imágenes). Esta etiqueta es obligatoria en cualquiera de los tipos de almacenamiento de la imagen propiamente dicha. Asimismo, la etiqueta *GainControl* posee un alto número de errores (536).

Hay que destacar que las etiquetas que muestran errores en el GPS (*GPSVersionID*, *GPSLatitude* y *GPSLongitude*) aparecen en el análisis como unas de las más problemáticas.

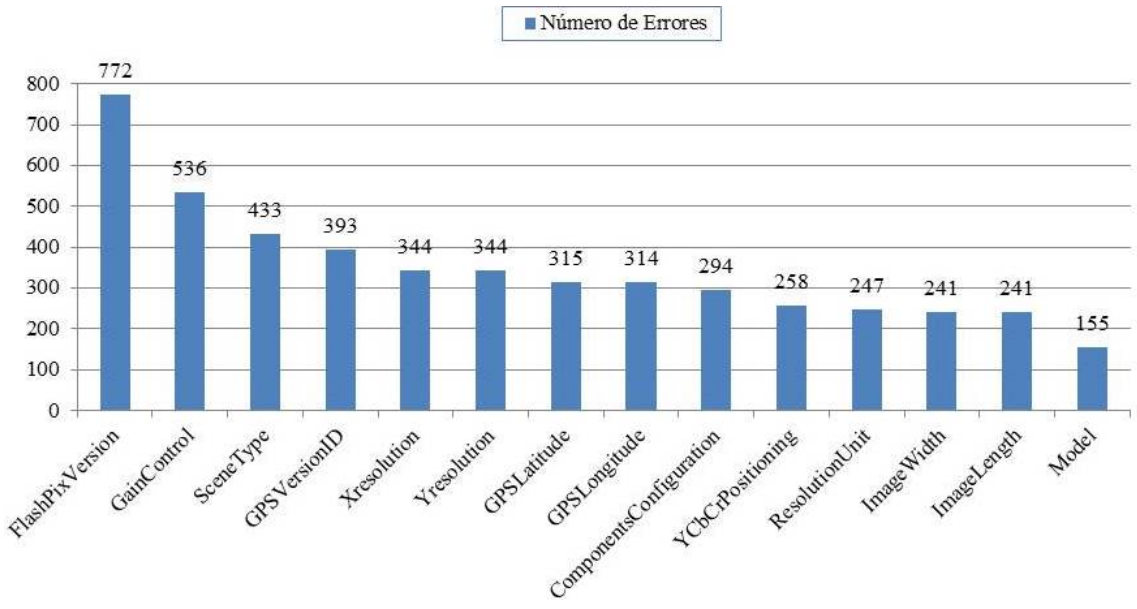


Figura 7.13: Etiquetas más propensas a tener errores

7.3 Síntesis del Capítulo

El objetivo de este capítulo ha sido realizar diversos análisis de metadatos a grandes bancos de imágenes utilizando *Theia*, centrándose especialmente en los metadatos de la especificación Exif, ya que es la más utilizada por los fabricantes de dispositivos móviles.

Como conclusión general se puede comentar que la inmensa mayoría de los fabricantes insertan información Exif en sus imágenes de gran utilidad, aún teniendo en cuenta que muchas veces no siguen fielmente la especificación o que los datos almacenados carecen de uniformidad.

Asimismo, se ha realizado una clasificación de 10 tipos de errores presentes en metadatos Exif mostrando ejemplos de estas anomalías detectadas.

Se han desarrollado diversos experimentos utilizando una base de datos de 4000 fotografías tomadas con 10 fabricantes de dispositivos móviles y varios de sus modelos más utilizados. La base de datos se compone de imágenes recolectadas directamente del dispositivo móvil y otras descargadas de bancos de imágenes como *Flickr* para analizar el origen de las anomalías. Estos análisis permiten concluir que muchos de los fabricantes no siguen la especificación Exif al 100 % indicando en el propio archivo lo contrario. Esto puede producir graves problemas en la extracción de los metadatos de las imágenes por medio de aplicaciones, así como problemas de interoperabilidad entre distintos dispositivos.

Theia incluye la funcionalidad de detectar los 10 tipos de errores Exif descritos y tratarlo tal y como se ha indicado anteriormente. Es fundamental que una herramienta de extracción de metadatos Exif tenga en cuenta este tipo de anomalías Exif a la hora de la extracción de los metadatos avisando al analista de la existencia de las mismas, ya que como se ha visto con los análisis estadísticos con gran probabilidad una imagen generada por un dispositivo móvil puede incluir uno o varios errores.

Finalmente, señalar que el análisis realizado se ha desarrollado sobre un banco con un universo de marcas y modelos de dispositivos móviles bastante heterogéneo y numeroso, por lo que los resultados obtenidos pueden ser tenidos en cuenta en futuros estudios.

Capítulo 8

Características del Ruido del Sensor y Transformada Wavelet

En este capítulo se desarrolla una técnica basada en las características del ruido y la transformada wavelet para la identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles. Se comienza con la exposición de unos conceptos generales. Seguidamente se detalla la técnica en sí. Previo a la presentación de los experimentos realizados, se indican los elementos utilizados con fines de clasificación para esta técnica. A continuación, se muestran los resultados de los experimentos sobre la identificación de la fuente de adquisición utilizando estas características. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

8.1 Generalidades

El proceso de generación de imágenes suele introducir varios defectos en éstas, que se traducen en ruido en la imagen final. Incluso si se toma una fotografía uniforme y completamente iluminada es posible observar pequeños cambios de intensidad entre los píxeles. Esto se debe al ruido de disparo que es aleatorio y, en gran parte, al patrón del ruido que es determinista y se mantiene aproximadamente igual si se toman varias fotografías de la misma escena. Un ruido de este tipo es causado por defectos de la matriz CFA, entre los que se incluyen píxeles calientes, píxeles muertos, defectos de columna y defectos de clúster. Este tipo de defectos provocan que dichos píxeles difieran en gran medida de los restantes de la imagen original, siendo en muchos casos indiferente que se tenga una u otra imagen, ya que este píxel muestra siempre el mismo valor. Por ejemplo, los píxeles muertos aparecen en la imagen como píxeles negros y los píxeles calientes aparecen como píxeles muy brillantes.

El patrón del ruido en una imagen se refiere a cualquier patrón espacial que no cambia de una imagen a otra. Se compone por el ruido espacial independiente de la señal o *Fixed Pattern Noise* (FPN) y por el ruido espacial debido a la diferencia de respuesta de cada píxel a la señal incidente o PRNU [KMC+06]. La estructura del patrón del ruido se ilustra en la Figura 8.1.

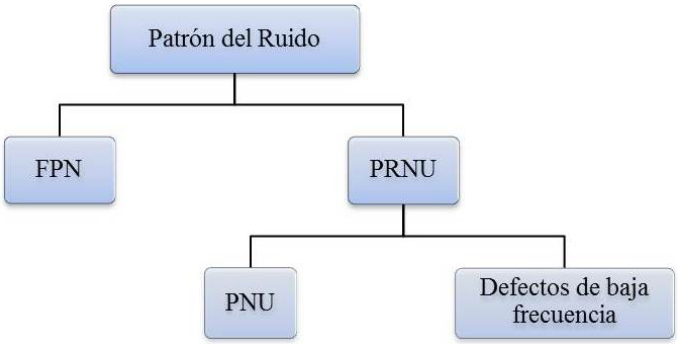


Figura 8.1: Patrón del ruido del sensor

El ruido FPN se genera por la corriente de oscuridad ("*dark current*") y depende de la exposición y de la temperatura. Debido a que el FPN es un ruido independiente aditivo, algunas cámaras lo eliminan automáticamente restando un marco oscuro a las imágenes que generan.

El ruido PRNU es la parte dominante en el patrón del ruido de las imágenes y es un ruido dependiente multiplicativo. El ruido PRNU está formado principalmente por la uniformidad de píxel (*Pixel Non-Uniformity* (PNU)) y por los defectos de baja frecuencia, como la configuración del *zoom* y la refracción de la luz en las partículas de polvo y en las lentes. El ruido PNU es la diferencia de sensibilidad a la luz entre los píxeles de la matriz del sensor. Se genera por la falta de homogeneidad de las obleas de silicio y las imperfecciones durante el proceso de fabricación del sensor. Debido a su naturaleza y origen es muy poco probable que incluso los sensores procedentes de la misma oblea presenten patrones PNU correlacionados. Este ruido no se ve afectado por la temperatura ambiente ni por la humedad. El ruido PNU es normalmente más común, complejo y significativo en los sensores de tipo CMOS debido a la complejidad de la circuitería de la matriz de píxeles.

8.2 Especificación de la Técnica

El objetivo de esta técnica es conseguir un conjunto de características basadas en el ruido que permita diferenciar la marca y el modelo de imágenes de dispositivos móviles. La extracción de las características del ruido se basa en [KMD09].

La extracción de las características del ruido se basa en [KMD09].

Sea I una imagen de $M * N$ píxeles, siendo M las filas y N las columnas. Si se denota por I_{ruido} al ruido correspondiente a la imagen original y por $I_{sinruido}$ a la imagen sin ruido, resulta que:

$$I_{ruido} = I - I_{sinruido} \tag{8.1}$$

Restando cada componente de color de la imagen sin ruido a cada componente de color de la imagen original, se obtienen componentes de ruido de cada píxel desglosados por componente de color.

El ruido de la imagen original I_{ruido} puede modelarse como la suma de dos componentes: el ruido constante $I_{ruidoconstante}$ y el ruido aleatorio $I_{ruidoaleatorio}$. Para los escáneres el ruido constante sólo depende del índice de la columna, ya que el mismo sensor es trasladado verticalmente para generar la imagen completa. La media del ruido de todas las columnas puede ser usada como patrón de referencia $\hat{I}_{ruidoconstante}(1, j)$, ya que las componentes aleatorias del ruido se anulan.

$$\hat{I}_{ruidoconstante}(1, j) = \frac{\sum_{i=1}^{M} I_{ruido}(i, j)}{M}, 1 \leq j \leq N \tag{8.2}$$

Para detectar la similitud entre las diferentes filas con el patrón de referencia se utiliza la correlación de éstas con dicho patrón:

$$correlation(X, Y) = \frac{(X - \overline{X}) \cdot (Y - \overline{Y})}{\|X - \overline{X}\| \cdot \|Y - \overline{Y}\|} \tag{8.3}$$

Posteriormente, se realiza el mismo proceso para detectar la similitud de las columnas con el patrón de referencia. Tras obtener las correlaciones entre las filas y entre las columnas se obtiene el conjunto de características en sí.

Para cada tipo de correlación se obtienen valores estadísticos de primer orden: media, mediana, máximo y mínimo. Se descarta como característica la moda, ya que después

de varios análisis y experimentos se ha observado como característica inútil, dado que al tratarse de valores flotantes no existen en la inmensa mayoría de los casos valores repetidos. Se han realizado pruebas truncando los valores flotantes, pero los resultados no han sido buenos, disminuyendo los porcentajes de acierto. Otras características de orden alto obtenidas son: varianza, *kurtosis* y *skewness*. Todas ellas miden valores estadísticos más específicos que las anteriores. Asimismo, se añaden las características del ratio entre las correlaciones de filas y de columnas. Por último, se incluye la característica del ruido medio por píxel.

En total se obtienen un conjunto de 16 características: 7 características de filas, 7 de columnas, el ratio entre las correlaciones de filas y de columnas y el ruido medio del píxel.

Además de las 16 características anteriores se utilizan otras basadas en la transformada wavelet. Cada banda de color se separa en tres sub-bandas, utilizando un nivel de descomposición QMF. Posteriormente, para cada banda de color se realiza la media de los valores obtenidos para cada una de las tres sub-bandas, obteniendo un total de 9 características basadas en la descomposición wavelet QMF. Finalmente, para la clasificación se utilizan un conjunto total de 25 características (16 basadas en el ruido y 9 basadas en la transformada wavelet). Para la clasificación se utiliza una máquina SVM con las características y configuración que se detalla en la siguiente sección.

8.3 Sistema de Clasificación

Los clasificadores SVM son ampliamente utilizados en la literatura. La elección del *kernel* depende, entre otros factores, de la naturaleza de los datos a clasificar. Se ha decidido utilizar un clasificador SVM con *kernel no lineal RBF*, pues no se dispone de información a priori de los datos: en los experimentos realizados no se impone ningún tipo de restricción sobre el contenido de la imagen. Los parámetros para la SVM son los siguientes: un kernel $\gamma = 2^3$ y un parámetro de coste $C = 32768$, es decir, los mismos que los utilizados en [RCAGSO+13]. Esta opción es la predominante en los trabajos más recientes del estado del arte y presenta buenos resultados. Existen muchas implementaciones de clasificadores SVM. En esta Tesis se ha optado por utilizar la librería LibSVM en la que SVM permite la clasificación de múltiples clases [CL].

Conviene recordar que la clasificación de imágenes que se realiza en este trabajo se hace sobre lo que puede denominarse un *conjunto cerrado* de elementos. Es decir, las clases de los elementos utilizados en el entrenamiento son las mismas clases que las utilizadas en la clasificación. Para todos los experimentos realizados las imágenes que se utilizan para la fase de entrenamiento no se utilizan para la fase de clasificación, lo cual hace a los experimentos más cercanos a escenarios realistas. Si se utilizan las mismas imágenes

para las fases de entrenamiento y clasificación, las tasas de acierto se incrementan notablemente, ya que la clasificación en enormemente más sencilla. Para todos los experimentos realizados en este trabajo el número de fotos utilizado por cada dispositivo móvil para el entrenamiento es el mismo que el utilizado para la clasificación.

En la implementación del código para la extracción de las características se utilizó inicialmente el lenguaje Python. Dados los problemas que se tuvieron por el excesivo tiempo de ejecución, se optó finalmente por migrar prácticamente todo a lenguaje C. La implementación en C no ha sido simplemente una mera traducción de Python, sino que se han optimizado los cálculos sobre los recorridos de las matrices de píxeles. Finalmente, estos módulos de código en C, han sido compilados para poder ser utilizados con el restante código en Python. De forma auxiliar se han utilizado las librerías Numpy, Scipy, PyWavelets y PIL del lenguaje Python.

Para seguir mejorando el rendimiento en términos de tiempo de ejecución, también se han implementado mecanismos de paralelización en la obtención de características, aprovechando los actuales sistemas multinúcleo. La paralelización en Python se ha realizado a nivel de proceso, es decir, se han hecho partes de ejecución en exclusión mutua que podían ejecutarse paralelamente. En cuanto al número de procesos a lanzar, depende del número de núcleos que tenga el computador en cuestión, aunque debido a la cantidad de memoria que requiere cada proceso (cuyo uso se incrementa masivamente según crece el tamaño de la imagen), en algunos casos conviene limitarlo a un número inferior a aquel.

8.4 Experimentos

La Tabla 8.1 muestra los dispositivos móviles utilizados para los experimentos y la configuración de las cámaras.

Tabla 8.1: Configuración utilizada en las cámaras de los dispositivos móviles

Marca	Modelo	Resolución	Condiciones de captura
Apple	iPhone 3G	2 MP (1600 × 1200)	Tipo de escena: Cualquiera Orientación: Vertical Flash: Deshabilitado Luz: Natural Balanceo de blancos: Auto Zoom Digital: 0 Tiempo de exposición: 0 Velocidad ISO: Automática
	iPhone 3GS	3,15 MP (2048 × 1536)	
Blackberry	8520	2 MP (1600 × 1200)	
Sony Ericsson	W580i	2 MP (1600 × 1200)	
	T707	3,15 MP (2048 × 1536)	
LG	Ku990i	5 MP (2592 × 1944)	
Nokia	5300	1,3 MP (1280 × 1024)	
	6110	2 MP (1600 × 1200)	
	N95	5 MP (2592 × 1944)	
	E61i	5 MP (2592 × 1944)	
HTC	Desire HD	8 MP (3264 × 2448)	

Los experimentos realizados tienen un doble objetivo: evaluar las características del ruido para la identificación de la fuente de adquisición y realizar un estudio sobre el número de imágenes a utilizar en la etapa de entrenamiento de la clasificación. El objetivo principal de la elección del número de imágenes en la clasificación es que los resultados sean lo más fiables posibles y que dependan en la menor medida posible del número de imágenes utilizadas. Hay que tener en cuenta que un número mayor no implica necesariamente una mejor clasificación. En este tipo de clasificadores, llegado un número de muestras en el entrenamiento, los resultados no mejoran con el incremento del número de muestras e incluso, en algunos casos, los resultados empeoran levemente. Por tanto, uno de los objetivos que se pretende es poder determinar para los experimentos un número óptimo de imágenes para la fase de entrenamiento.

En el primer experimento se utilizan 50 imágenes de 10 dispositivos móviles (Iphone 3G, Iphone 3GS, Blackberry 8520, HTC Desire HD, LG Ku990i, Nokia 5300, Nokia 6110, Nokia N95, Nokia E61i, Sony-Ericsson W580i), obteniendo una tasa media de acierto del 89,45 %. La primera conclusión de esta prueba es que aún con un número de imágenes reducido los resultados son buenos, ya que el número de dispositivos es alto.

En el segundo experimento se utilizan 7 dispositivos móviles y entre 150 y 200 imágenes por dispositivo móvil. Los dispositivos utilizados son: Blackberry 8520, HTC Desire HD, LG Ku990i, Nokia 5300, Nokia 6110, Nokia 6300, Sony-Ericsson T707 y Sony-Ericsson W580i. La tasa media de acierto para este experimento es del 94,22 %. Con un mayor número de imágenes los resultados parecen mejorar, si bien hay que tener en cuenta que hay 3 dispositivos móviles menos.

Con el objetivo de hacer una comparación más directa sin que afecte a los resultados el número de dispositivos utilizados se realizan dos experimentos con 50 y 150 imágenes utilizando 5 dispositivos móviles. Los dispositivos utilizados son: Blackberry 8520, HTC Desire HD,LG Ku990i, Nokia 5300 y Sony Ericsson W580i. Los resultados son del 95,86 % y 96,2 % para 50 y 150 imágenes, respectivamente. Como puede observarse, la diferencia de la tasa de acierto es mínima, corroborando que las tasas de acierto son prácticamente invariables a partir de un número de imágenes. Asimismo, puede deducirse algo, que en cierta forma tiene más sentido común, en comparación con los resultados de los dos primeros experimentos: a menor número de dispositivos mayor es la tasa de acierto en la identificación de la fuente de adquisición.

En el último experimento se prueban con distintos tramos de números de imágenes utilizando sólo dos dispositivos: el Blackberry 8520 y el Sony-Ericsson W580i. El objetivo es obtener con mayor precisión datos empíricos que permitan decidir cuál va a ser el número de imágenes que se van a utilizar en los restantes experimentos.

La Tabla 8.2 muestra que la tasa de acierto no varía significativamente, por lo que se puede concluir que el número de imágenes no afecta significativamente para la clasificación basada en las características del contenido.

Tabla 8.2: Tasa media de acierto por número de imágenes

Número de imágenes	Tasa media de acierto
30	96,67 %
60	95 %
90	94,44 %
120	96,67 %
150	96,33 %

Tras este último experimento y teniendo en cuenta los datos de las referencias del estado del arte, se estima que un número de imágenes que cumpla las expectativas previstas puede ser 100. Esta elección se ha realizado teniendo en cuenta que cualquier otro número dentro del rango 30-150 podría ser bueno, dada la estabilidad de los resultados. Aún así, se estima que menos de 100 imágenes pueden ser insuficientes, sobre todo teniendo en cuenta los experimentos en los que el número de dispositivos es pequeño.

En relación con los experimentos realizados también se concluye que la técnica obtiene buenos resultados, ya que la media de la tasa media de acierto para todos los experimentos es del 94,98 %. Hay que destacar que, ante las variadas situaciones de aplicación de la técnica como puede ser la variación en el número de imágenes y de dispositivos, los resultados se han mantenido estables (una diferencia en la tasa media de acierto del 7,22 % en el peor de los casos).

8.5 Síntesis del Capítulo

El objetivo de este capítulo ha sido presentar la técnica de identificación de la fuente de adquisición de imágenes digitales basadas en las características del ruido del sensor y en la transformada wavelet. En primer lugar se han explicado las 25 características que utiliza el algoritmo. Luego se han descrito los elementos utilizados para la clasificación. Posteriormente, se ha estudiado el número de imágenes a utilizar en la fase de entrenamiento para el clasificador SVM, concluyéndose que 100 imágenes es un buen número. Finalmente, se han realizado otros experimentos para evaluar la validez de la técnica.

Capítulo 9

Patrón de No-Uniformidad de la Foto-Respuesta del Sensor

Este capítulo proporciona una técnica de identificación de la fuente de adquisición de imágenes digitales basada en el patrón de No-Uniformidad de la Foto-Respuesta del Sensor. Se comienza con una especificación detallada de la técnica. Posteriormente, se presentan los experimentos realizados utilizando distintas opciones de configuración que posee el algoritmo. El capítulo finaliza con una breve síntesis de lo expuesto en el mismo.

9.1 Generalidades

La literatura muestra que el ruido SPN [GBK+01] [LFG06] [Li09] y la transformada wavelet [MKY08] [OA11] son métodos efectivos para la identificación de la fuente de adquisición de una imagen. Sin embargo, la mayoría de los trabajos están centrados en DSCs, excluyendo las cámaras de dispositivos móviles.

Debido a la propiedad determinista del patrón del ruido del sensor que está presente en una imagen, se puede usar este patrón como huella para identificar el dispositivo que la genera. Haciendo una analogía, se puede decir que el patrón del ruido del sensor es para una cámara digital lo que la huella dactilar para un ser humano. El enfoque que se presenta caracteriza la huella utilizando un vector de características basadas en la Transformada Wavelet. El esquema de la Figura 9.1 contiene el diagrama funcional de la propuesta.

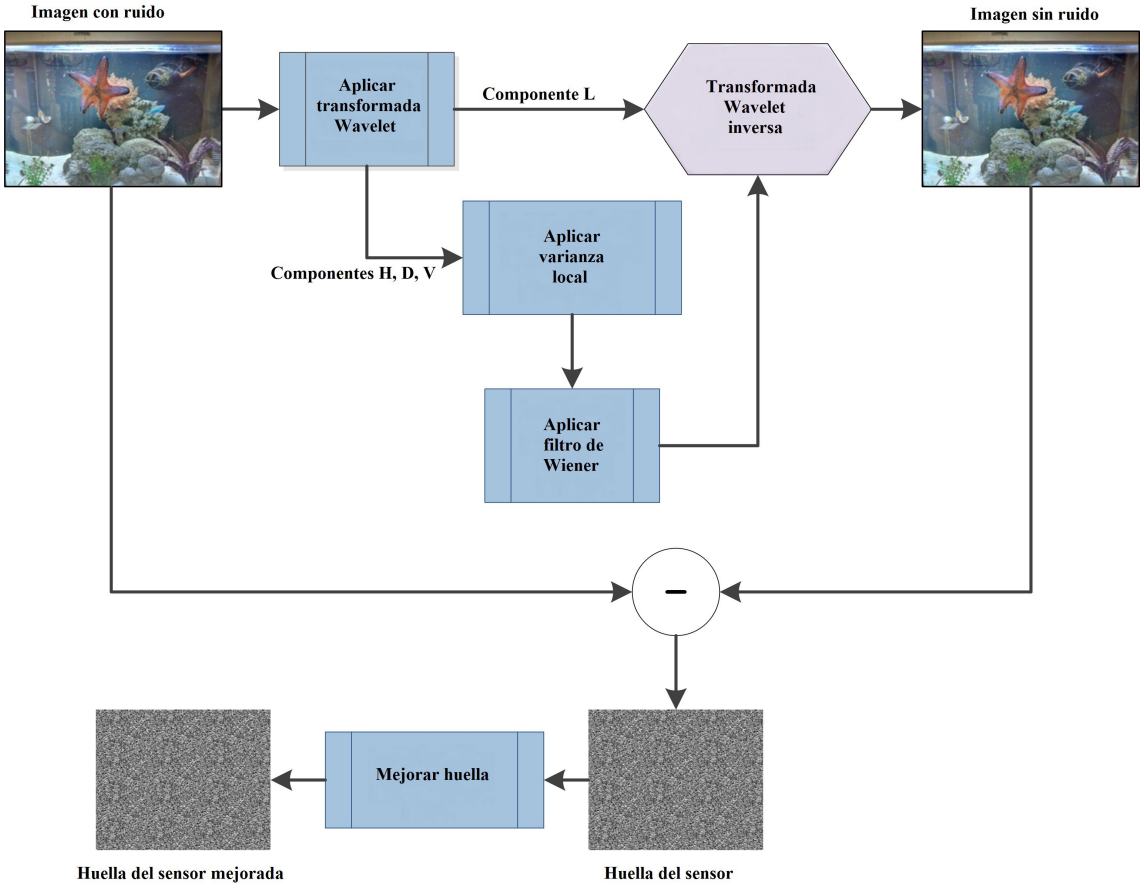


Figura 9.1: Esquema funcional de extracción del patrón del ruido del sensor

9.2 Especificación de la Técnica

Para poder identificar la fuente de adquisición se requiere de un algoritmo que permita extraer el ruido del sensor y otro que permita obtener las características de las huellas obtenidas, para así poder clasificarlas e identificarlas.

Tomando como referencia las ideas principales de [LFG06] se propone el algoritmo 1 para extraer el ruido del sensor.

Algoritmo 1: Extracción del ruido del sensor

Input: Imagen
varianza: (adaptativa o no adaptativa)
Result: Huella del sensor de la imagen

1 **procedure** ExtraerHuella(I)
2 Realizar descomposición wavelet de 4 niveles de I_{limpia};
3 **foreach** *nivel de la descomposición wavelet* **do**
4 **foreach** $c \in \{H,V,D\}$ **do**
5 Calcular la varianza local;
6 **if** *varianza adaptativa* **then**
7 Calcular 4 varianzas con ventanas de tamaños 3, 5 ,7 y 9;
8 Seleccionar la varianza mínima;
else
9 Calcular la varianza con una ventana de tamaño 3;
10 Calcular los componentes wavelet sin ruido aplicando el filtro de Wiener a la varianza;
11 Obtener la imagen limpia del ruido del sensor aplicando la Transformada Wavelet Inversa;
12 Calcular el ruido del sensor con $I_{ruido}=I_{entrada}-I_{limpia}$;
13 Aplicar a I_{ruido} un promediado a cero;
14 Aumentar el peso del canal verde con $I_{ruido}=0{,}3\cdot I_{ruido_R}+0{,}6\cdot I_{ruido_G}+0{,}1\cdot I_{ruido_B}$;
15 **end procedure**

A continuación se describen detalladamente los pasos del mismo:

En el paso 1 del algoritmo se descompone la imagen en sus canales de color R, G y B. Para cada canal de color se realiza una descomposición wavelet en cuatro niveles de la imagen utilizando QMF Daubechies 8-tap. El número de niveles de la descomposición puede variarse modificando de esta manera el procesamiento requerido y también la calidad de la información extraída de las imágenes.

Una vez aplicada la descomposición wavelet de la imagen, para cada nivel de la descomposición se obtienen los componentes de alta frecuencia H, V y D. Se denota a los componentes de alta frecuencia como $c(i,j)$ donde $c \in \{H,V,D\}$ y donde (i,j) corre a través de un conjunto de índices J que depende del nivel de la descomposición.

Para cada componente de alta frecuencia se calcula la varianza local usando la estimación *Maximum A-Posteriori Probability* (MAP). La información estadística de las imágenes puede diferir mucho entre ellas e incluso entre las diferentes regiones de una misma imagen. Para combatir esta diferencia se utilizan estimadores adaptativos con diferentes tamaños de ventana de vecindad.

En el algoritmo propuesto se plantea calcular la varianza local de forma adaptativa o no, dependiendo de las necesidades y/o restricciones computacionales.

En el caso de requerir estimadores adaptativos, para cada componente de alta frecuencia se calcula la varianza local para cuatro tamaños de ventana $W \times W$ de vecindario N con $W \in \{3, 5, 7, 9\}$. El parámetro σ_0, que controla qué tan fuerte será la supresión de ruido, toma el valor de 5 (ampliamente recomendado en la literatura).

$$\hat{\sigma}^2(i,j) = \text{máx}\left(0, \frac{1}{W^2} \sum_{(i,j)\in N} c^2(i,j) - \sigma_0^2\right), \quad (i,j) \in J \tag{9.1}$$

A continuación, se toma el mínimo de las 4 varianzas como la estimación final:

$$\hat{\sigma}^2(i,j) = \text{mín}\left(\sigma_3^2(i,j), \sigma_5^2(i,j), \sigma_7^2(i,j), \sigma_9^2(i,j)\right), \quad (i,j) \in J \tag{9.2}$$

En el caso de no requerir la utilización de estimadores adaptativos que requieren de más procesamiento se calcula la varianza local para un solo tamaño de ventana considerando $W = 3$.

Después, se calculan los componentes wavelet sin ruido utilizando el filtro de Wiener:

$$c_{limpio}(i,j) = \quad c(i,j) \quad \frac{\hat{\sigma}^2(i,j)}{\hat{\sigma}^2(i,j) + \sigma_0^2} \tag{9.3}$$

Por último, la imagen final sin ruido se obtiene aplicado la transformada wavelet inversa a los componentes wavelet sin ruido de cada uno de los niveles.

Teniendo la imagen sin ruido se obtiene la huella del sensor.

En el paso 2 se calcula la huella o ruido residual I_{ruido} de una imagen I eliminando el contenido de la escena de la imagen mediante un filtro de eliminación de ruido F.

$$I_{ruido} = \quad I \quad - \quad F(I) \tag{9.4}$$

Entre los diferentes filtros que existen para la eliminación del ruido de las imágenes los que usan la transformada wavelet dan mejor resultado, debido a que el ruido residual que se obtiene con este filtro contiene la menor cantidad de rasgos de la escena. Generalmente, las áreas que están alrededor de los bordes son mal interpretadas cuando se utilizan únicamente filtros de eliminación de ruido menos robustos, tales como el filtro de Wiener o el filtro de mediana. Por este motivo se selecciona el filtro de eliminación de ruido basado en la transformada wavelet. La huella calculada I_{ruido} contiene todos los elementos que se

presentan sistemáticamente en cada una de las imágenes, incluyendo algunos que no son causados por el sensor como las características causadas por la interpolación de colores o la compresión JPEG. La mayoría de estas características son generadas por el proceso de interpolación cromática dependiendo de la CFA utilizada por la cámara. Debido a que estas características tienen una naturaleza periódica se pueden eliminar mediante el paso 3.

En el paso 3 del algoritmo se limpia la huella de las características que no son intrínsecas al sensor aplicando un promediado a cero (también denominado *zero mean*) de filas y columnas como se sugiere en [CFGL08], de tal manera que los promedios de las filas y de las columnas sean iguales a cero. Esto se logra restando el promedio de la columna a cada píxel de la columna y posteriormente restando el promedio de la fila a cada píxel de la fila. Esta operación se aplica a todas las filas y columnas de la imagen.

En el paso 4 se da mayor peso al canal verde ya que éste, debido a la configuración de la matriz de color, contiene más información sobre la imagen que el resto de los canales de color [APS98] [McK07] [CSA08]:

$$I_{ruido} = \quad 0,3 \cdot I_{ruido_R} + 0,6 \cdot I_{ruido_G} + 0,1 \cdot I_{ruido_B} \tag{9.5}$$

Una vez extraída la huella del sensor de la imagen, se calculan un total de 81 características (3 canales x 3 componentes wavelet x 9 momentos centrales) mediante el algoritmo 2.

Algoritmo 2: Extracción de características

Input: Imagen
Huella del sensor de la imagen
Result: 81 características

1. **procedure** ExtraerCaracteristicas(I)
2. Separar los canales R, G y B de la huella del sensor;
3. **foreach** *canal de color* **do**
4. Hacer una descomposición wavelet de un nivel;
5. **foreach** $c \in \{H,V,D\}$ **do**
6. Calcular k momentos centrales con $m_k = \frac{1}{n}\sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7. **end procedure**

En este algoritmo los momentos centrales de los valores absolutos de cada componente de alto nivel de la huella del sensor se toman como una medida de la distribución del ruido.

En el paso 1 se separa la huella del sensor en los 3 canales de color R, G y B.

A continuación, en el paso 2 se realiza una descomposición wavelet con 8-tap Daubechies para obtener los componentes de alta frecuencia H, V y D de los que se extraen las características.

En el paso 3 para cada componente wavelet c_i se calcula el k-ésimo momento central absoluto utilizando la ecuación 9.6 para $k = 9$.

$$m_k = \frac{1}{n} \sum_{i=1}^{n} |c_i - \overline{c}|^k \tag{9.6}$$

Para cada componente wavelet H, V y D de cada canal de color se calculan 9 momentos, obteniéndose un total de $3 \times 3 \times 9 = 81$ características como se ilustra en la Tabla 9.1.

Tabla 9.1: Momentos centrales característicos de la huella del sensor

Canal de color	Componente wavelet	Momentos centrales					
R	H	m_1	m_2	m_3	m_4	...	m_9
	V	m_1	m_2	m_3	m_4	...	m_9
	D	m_1	m_2	m_3	m_4	...	m_9
G	H	m_1	m_2	m_3	m_4	...	m_9
	V	m_1	m_2	m_3	m_4	...	m_9
	D	m_1	m_2	m_3	m_4	...	m_9
B	H	m_1	m_2	m_3	m_4	...	m_9
	V	m_1	m_2	m_3	m_4	...	m_9
	D	m_1	m_2	m_3	m_4	...	m_9

9.3 Experimentos

Antes de la descripción de los experimentos en sí, se van a presentar unos tiempos de ejecución orientativos de los distintos algoritmos y clasificadores utilizados. El algoritmo de extracción de la huella y la extracción de las características han sido implementados en el lenguaje de programación Python 2.7 y en el lenguaje C. En un Intel Core i7 Q720 1.6 GHz con 8GB de *Random-Access Memory* (RAM) la extracción de las características de un recorte de 1024 × 1024 de una imagen, utilizando la estimación de la varianza adaptativa y el promediado a cero, tarda 20 s aproximadamente. Con la misma configuración para un recorte de 512 × 512 tarda aproximadamente 5 s. El mismo caso pero no utilizando la estimación de la varianza adaptativa tarda 5 s y 1,5 s para los recortes de 1024 × 1024 y 512 × 512, respectivamente. La fase de entrenamiento y de clasificación con la máquina SVM para 600 imágenes se realiza en un minuto y una fracción de segundo, respectivamente. Sin embargo, para la ejecución de los experimentos se ha utilizado EOLO como recurso de computación.

En la Tabla 9.2 se muestran las marcas y los modelos de los móviles utilizados en los experimentos, así como las características de la cámara y la configuración básica de la misma en el momento de la toma.

Tabla 9.2: Configuración utilizada en las cámaras de los dispositivos móviles

Marca	Modelo	Resolución	Condiciones de captura
Apple	iPhone3G (A1)	2 MP (1600 × 1200)	Tipo de escena: Cualquiera Orientación: Vertical Flash: Deshabilitado Luz: Natural Balanceo de blancos: Auto Tasa de zoom digital: 0 Tiempo de exposición: 0 seg Velocidad ISO: Automático
	iPhone4S (A2)	8 MP (3264 × 2448)	
	iPhone3 (A3)	2 MP (1600 × 1200)	
	iPhone5 (A4)	8 MP (3264 × 2448)	
Black Berry	8520 (B1)	2 MP (1600 × 1200)	
Sony Ericsson	UST25a (SE1)	5 MP (2592 × 1944)	
	U5I (SE2)	8 MP (3264x2448)	
Samsung	GT-I9100 (S1)	8 MP (3264 × 2448)	
	GT-S5830 (S2)	5 MP (2592 × 1944)	
	GT-S5830M (S3)	5 MP (2592 × 1944)	
	EK-GC101 (S4)	16,3 MP (4608 × 3456)	
LG	E400 (L1)	3,2 MP (2048 × 1536)	
	P760 (L2)	5 MP (2592 × 1944)	
HTC	Desire HD (H1)	8 MP (3264 × 2448)	
	Desire (H2)	5 MP (2592 × 1944)	
Nokia	E61I (N1)	2 MP (1600 × 1200)	
	800-Lumia (N2)	8 MP (3264 × 2448)	
Zopo	ZP979 (Z1)	12,6 MP (4096 × 3072)	

Para comprobar la efectividad de los algoritmos propuestos se ha realizado un conjunto de experimentos variando todos los posibles parámetros de configuración. La Tabla 9.3 resume los distintos parámetros de configuración y sus posibles valores.

Tabla 9.3: Parámetros utilizados en el algoritmo propuesto y sus posibles valores

Parámetro	Posibles Valores
Número de imágenes para el entrenamiento por cámara	100
Número de imágenes para la clasificación por cámara	100
Recorte de la imagen	Centrado (1024 × 1024 o 512 × 512)
Estimación de la varianza	Adaptativa (pasos 7 y 8 del Algoritmo 1) o No adaptativa (paso 9 del Algoritmo 1)
Promediado a cero	Utilizado o no utilizado (paso 13 del Algoritmo 1)

Previamente, se había realizado un experimento con el objetivo de probar la estabilidad de los resultados de la técnica con respecto a experimentos iguales salvo en la utilización de distintas imágenes. En este experimento se usaron un grupo de 8 dispositivos móviles de 4 fabricantes diferentes. Se consideraron de Apple los modelos iPhone3G e iPhone4S, de BlackBerry el 8520, de Sony Ericsson el UST25a y el U5I, y de Samsung el GTI9100 y el GTS5830.

Se han realizado 10 experimentos, utilizando 10 conjuntos diferentes de 100 imágenes por cámara, calculando para cada experimento la tasa media de acierto en la clasificación. La tasa media de acierto cambió ligeramente en cada una de las ejecuciones lo que indica que existe estabilidad en los resultados sobre los distintos conjuntos de imágenes para el entrenamiento y la clasificación.

La matriz de confusión del caso mejor, medio y peor se muestran, respectivamente, en las Tablas 9.4, 9.5 y 9.6. La tasa media de acierto para la identificación de la fuente de adquisición fue del 93,2 %.

Tabla 9.4: Matriz de confusión del mejor resultado (93,87 %)

Cámara	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**96**	1	0	0	0	0	0	3
A2	0	**97**	0	0	0	0	3	0
A3	0	0	**98**	0	0	0	2	0
B1	0	0	0	**94**	0	4	0	2
SE1	11	1	0	0	**88**	0	0	0
SE2	3	0	0	1	0	**93**	1	2
S1	4	8	0	0	0	3	**85**	0
S2	0	0	0	0	0	0	0	**100**

Tabla 9.5: Matriz de confusión del resultado medio (93,25 %)

Cámara	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**94**	1	0	0	0	1	0	4
A2	0	**96**	0	0	1	0	3	0
A3	0	0	**97**	0	0	0	2	1
B1	0	0	0	**94**	0	2	0	4
SE1	10	1	0	0	**89**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	5	6	0	0	0	6	**83**	0
S2	0	0	0	0	0	1	0	**99**

Tabla 9.6: Matriz de confusión del peor resultado (92,62 %)

Cámara	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**92**	1	0	0	0	0	0	7
A2	0	**96**	0	0	1	0	3	0
A3	0	1	**99**	0	0	0	0	0
B1	0	0	3	**91**	0	4	0	2
SE1	7	2	0	0	**91**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	4	10	0	0	0	7	**79**	0
S2	0	0	0	0	0	1	0	**99**

En la Tabla 9.7 se muestra el conjunto de experimentos realizados con sus correspondientes parámetros de configuración. Con respecto a los tamaños y lugares de donde extraer el recorte, se consideran los píxeles centrales de la imagen tal y como se recomienda en [LS11]. Asimismo, se recomienda un tamaño de recorte de 1024 × 1024 [LS11].

Tabla 9.7: Parámetros de configuración de los experimentos

Experimento	Resolución	Número de dispositivos	Varianza Adaptativa	Promediado a Cero	Tasa media de acierto
Experimento 1	1024 × 1024	6	Sí	Sí	96,33 %
Experimento 2	1024 × 1024	6	Sí	No	98 %
Experimento 3	1024 × 1024	6	No	Sí	97,5 %
Experimento 4	1024 × 1024	6	No	No	97,83 %
Experimento 5	512 × 512	6	Sí	Sí	73,76 %
Experimento 6	512 × 512	6	Sí	No	93,17 %
Experimento 7	512 × 512	6	No	Sí	92,5 %
Experimento 8	512 × 512	6	No	No	91,67 %
Experimento 9	1024 × 1024	14	No	No	87,21 %

9.3.1 Experimento 1

Los parámetros para este experimento son: recorte centrado de 1024 × 1024, estimación de la varianza adaptativa y promediado a cero. En la Tabla 9.8 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 96,33 %.

Tabla 9.8: Experimento 1

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**96**	0	2	0	2	0
Samsung EK-GC101	5	**88**	2	2	3	0
Nokia 800-Lumia	0	0	**100**	0	0	0
Zopo ZP979	0	0	2	**98**	0	0
LG P760	0	0	0	0	**100**	0
Sony Ericsson ST25A	0	0	3	0	1	**96**

9.3.2 Experimento 2

Los parámetros para este experimento son: recorte centrado de 1024 × 1024, estimación de la varianza adaptativa y no promediado a cero. Es decir, los mismos parámetros que en el

experimento 1, salvo que en este experimento no se aplica el promediado a cero. Dado que las imágenes utilizadas para todos los experimentos son las mismas se puede comprobar cómo afecta este cambio.

En la Tabla 9.9 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 98 %.

Tabla 9.9: Experimento 2

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**97**	0	1	0	2	0
Samsung EK-GC101	1	**95**	0	3	1	0
Nokia 800-Lumia	0	0	**100**	0	0	0
Zopo ZP979	0	0	2	**98**	0	0
LG P760	0	0	0	0	**99**	1
Sony Ericsson ST25A	0	0	0	0	1	**99**

Se observa que el promediado a cero empeora la tasa media de acierto (1,67 % con respecto al experimento 1), aunque la diferencia no es muy significativa y hay que esperar a los resultados de los siguientes experimentos para poder sacar conclusiones definitivas. Asimismo, puede observarse que, a excepción del modelo LG P760 (que pasa de un 100 % a un 99 %), para todos los dispositivos móviles la tasa media de acierto aumenta.

9.3.3 Experimento 3

Los parámetros para este experimento son: recorte centrado 1024 × 1024, estimación de la varianza no adaptativa y promediado a cero. Es decir, los mismos parámetros que en el experimento 1, salvo que en este experimento se aplica varianza no adaptativa. El objetivo de este experimento es comprobar si el tipo de varianza elegido es determinante en los resultados del algoritmo. Asimismo, es importante destacar que el uso de la varianza adaptativa o no adaptativa tiene efectos relevantes en el tiempo de ejecución del algoritmo, ya que su ejecución con varianza no adaptativa es aproximadamente del orden de 4 veces más rápida.

En la Tabla 9.10 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 97,5 %.

Tabla 9.10: Experimento 3

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**95**	0	2	0	3	0
Samsung EK-GC101	1	**95**	0	3	1	0
Nokia 800-Lumia	0	0	**100**	0	0	0
Zopo ZP979	0	1	1	**98**	0	0
LG P760	0	1	0	0	**99**	1
Sony Ericsson ST25A	0	0	1	0	1	**98**

Se esperaba que la estimación de varianza no adaptativa obtuviera peores tasas de acierto, pero una vez vistos los resultados de los experimentos anteriores, se puede observar que la tasas de acierto no difieren mucho.

9.3.4 Experimento 4

Los parámetros para este experimento son: recorte centrado de 1024 × 1024, estimación de varianza no adaptativa y no promediado a cero. Es decir, los mismos parámetros que en el experimento 2, salvo que en este experimento se aplica varianza no adaptativa. Al igual que en el experimento anterior el objetivo de este experimento es comprobar si el tipo de varianza elegida es determinante en los resultados del algoritmo para un tamaño de recorte menor. Además, se busca observar el comportamiento del promediado a cero para varianza no adaptativa.

En la Tabla 9.11 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 97,83 %.

Tabla 9.11: Experimento 4

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**96**	2	0	0	2	0
Samsung EK-GC101	1	**95**	0	3	1	0
Nokia 800-Lumia	0	0	**100**	0	0	0
Zopo ZP979	0	0	2	**98**	0	0
LG P760	0	2	0	0	**99**	0
Sony Ericsson ST25A	0	0	1	0	0	**100**

Al contrario que ocurre en los experimentos 1 y 3, en éste se observa un pequeño empeoramiento sobre la tasa media de acierto del experimento 2. Para el caso del tamaño de recorte 1024 × 1024 se concluye según estos experimentos que el uso de la varianza adaptativa no mejora notablemente los resultados, ya que prácticamente los resultados son

los mismos con pequeñas mejoras o empeoramientos. Además la diferencia de resultados con respecto al experimento 3 es de una mejora insignificante por lo que en principio se puede concluir que el uso del promediado a cero combinado con la varianza no adaptativa no aporta mejoras notables.

9.3.5 Experimento 5

Los parámetros para este experimento son: recorte centrado de 512 × 512, estimación de varianza adaptativa y promediado a cero. Es decir, los mismos parámetros que en el experimento 1, salvo que en este experimento se reduce el tamaño del recorte. El objetivo de este experimento y los tres siguientes es comprobar la influencia que tiene el tamaño del recorte en los resultados.

En la Tabla 9.12 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 73,67 %.

Tabla 9.12: Experimento 5

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**93**	3	2	0	2	0
Samsung EK-GC101	16	**76**	2	0	6	0
Nokia 800-Lumia	0	0	**86**	0	2	12
Zopo ZP979	0	19	2	**0**	79	0
LG P760	2	0	1	0	**93**	4
Sony Ericsson ST25A	0	0	4	0	2	**94**

Como era de esperar la tasa media de acierto baja considerablemente (un 7 %) con respecto al experimento 1, ya que la cantidad de información de la imagen utilizada para obtener las características es considerablemente menor.

9.3.6 Experimento 6

Los parámetros para este experimento son: recorte centrado de 512 × 512, estimación de varianza adaptativa y no promediado a cero. Es decir, los mismos parámetros que en el experimento 5, salvo que en este experimento no se utiliza el promediado a cero. Este experimento tiene entre otros el objetivo de ver la influencia del promediado a cero utilizando tamaños de recorte pequeños con varianza adaptativa.

En la Tabla 9.13 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 93,17 %.

Tabla 9.13: Experimento 6

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**94**	2	3	0	1	0
Samsung EK-GC101	6	**91**	1	1	1	0
Nokia 800-Lumia	0	0	**93**	0	0	7
Zopo ZP979	0	5	2	**92**	1	0
LG P760	2	0	0	0	**95**	3
Sony Ericsson ST25A	0	0	4	0	2	**94**

Como era de esperar la tasa media de acierto baja (4,83 %) con respecto al experimento 2 debido a la reducción del tamaño del recorte. Para el caso de un tamaño de recorte menor el no utilizar promediado a cero hace que la pérdida de porcentaje de acierto sea menor que en el caso del experimento anterior (3,84 %), aunque este cambio no es una mejora significativa.

9.3.7 Experimento 7

Los parámetros para este experimento son: recorte centrado 512 × 512, estimación de la varianza no adaptativa y promediado a cero. Es decir, los mismos parámetros que en el experimento 5, salvo que en este experimento se usa varianza no adaptativa. El objetivo de este experimento, entre otros, es ver la influencia de la varianza adaptativa utilizando tamaños de recorte pequeños.

En la Tabla 9.14 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 92,50 %.

Tabla 9.14: Experimento 7

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**95**	0	2	0	3	0
Samsung EK-GC101	5	**89**	0	3	3	0
Nokia 800-Lumia	0	0	**85**	0	1	14
Zopo ZP979	0	1	2	**97**	0	0
LG P760	2	0	2	0	**93**	3
Sony Ericsson ST25A	0	0	4	0	0	**96**

Como era de esperar la tasa media de acierto baja (un 5 %) con respecto al experimento 3 debido a la reducción del tamaño del recorte. Con respecto a la comparativa con el experimento 5, se puede observar que los resultados con varianza no adaptativa son mucho mejores (3,17 %). Sin embargo, con el experimento 6, que también utiliza varianza

adaptativa, la variación de resultados es mínima (0,67 %).

9.3.8 Experimento 8

Los parámetros para este experimento son: recorte centrado de 512 × 512, estimación de la varianza no adaptativa y no promediado a cero. Es decir, los mismos parámetros que en el experimento 4, salvo que en este experimento se reduce el tamaño del recorte. El objetivo de este experimento, entre otros, es ver la influencia del promediado a cero utilizando tamaños de recorte pequeños y varianza no adaptativa.

En la Tabla 9.15 se muestra la matriz de confusión para este experimento. La tasa media de acierto para la identificación de la fuente de adquisición es del 91,67 %.

Tabla 9.15: Experimento 8

Cámara	Apple iPhone5	Samsung EK-GC101	Nokia 800-Lumia	Zopo ZP979	LG P760	Sony Ericsson ST25A
Apple iPhone5	**95**	0	2	0	3	0
Samsung EK-GC101	5	**89**	0	3	3	0
Nokia 800-Lumia	0	0	**85**	0	1	14
Zopo ZP979	0	1	2	**97**	0	0
LG P760	2	0	2	0	**93**	3
Sony Ericsson ST25A	0	0	4	0	0	**96**

Como era de esperar la tasa media de acierto baja (un 6,16 %) con respecto al experimento 4 debido a la reducción del tamaño del recorte. Se constata los resultados obtenidos entre la comparación de los resultados del experimento 6 y 8, es decir, que el uso de la varianza adaptativa no mejora notablemente los resultados.

9.3.9 Experimento 9

Con el objetivo de evaluar la escalabilidad de la técnica respecto al número de dispositivos, se realiza un experimento con un grupo de 14 dispositivos móviles de 7 fabricantes distintos. Como puede verse en la matriz de confusión de la Tabla 9.16 la tasa media de acierto cayó al 87,21 %. Claramente puede verse que hay un pequeño descenso de la tasa de acierto a medida de que el número de cámaras (clases) se incrementa.

Tabla 9.16: Experimento 9

Cámara	A1	A2	A3	A4	B1	SE1	SE1	S1	S1	S3	L1	H1	H2	N1
A1	**90**	0	0	2	0	0	0	0	7	0	1	0	0	0
A2	0	**91**	0	3	0	0	0	3	0	0	0	1	2	0
A3	0	0	**98**	0	0	0	0	2	0	0	0	0	0	0
A4	0	0	1	**88**	0	0	0	0	0	0	3	6	0	2
B1	0	0	0	2	**73**	0	0	0	4	0	0	1	0	20
SE1	7	0	0	0	0	**80**	0	0	0	0	1	12	0	0
SE2	1	0	0	2	2	0	**86**	1	2	5	1	0	0	0
S1	4	5	0	4	0	0	1	**83**	0	0	1	0	2	0
S2	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0
S3	0	0	1	0	0	0	8	0	0	**85**	0	1	0	5
L1	0	0	0	9	0	6	0	0	2	0	**70**	13	0	0
H1	2	0	0	0	0	11	0	0	1	0	1	**85**	0	0
H2	0	6	0	0	0	0	0	0	0	0	0	0	**94**	0
N1	0	0	0	0	2	0	0	0	0	0	0	0	0	**98**

9.4 Síntesis del Capítulo

En este capítulo se han presentado la técnica de identificación de la fuente de adquisición basada en la No-Uniformidad de la Foto-Respuesta del Sensor. Concretamente, el algoritmo de extracción obtiene 81 características. Asimismo, se han realizado un conjunto de experimentos con el objetivo de obtener conclusiones para la evaluación de la técnica teniendo en cuenta las distintas opciones de configuración.

La primera conclusión es que, independientemente de los parámetros utilizados en el algoritmo, se obtienen peores resultados cuanto menor es el tamaño de recorte utilizado. No existe ningún caso en el que la tasa media de acierto aumente con la disminución del tamaño de recorte. Obviamente, en términos de tiempo de ejecución, el proceso demora más cuanto mayor es el tamaño de recorte utilizado.

La segunda conclusión general es que no existen claramente definidos unos parámetros de configuración del algoritmo para cada tamaño de recorte que permita obtener los mejores resultados. Cualquier combinación de parámetros obtiene unos resultados similares, aunque hay que destacar que hay parámetros que optimizan la tasa de acierto en mayor medida. Es responsabilidad del analista forense la utilización de los parámetros de configuración que logren mejores resultados a costa de mayores tiempos de ejecución o lo contrario. También se concluye que ninguno de los parámetros utilizados es superfluo ya que ninguno de ellos independientemente empeora los resultados para todas las posibles combinaciones.

La tercera conclusión general, es que tanto para recortes grandes como pequeños existe una configuración que obtiene los mejores resultados: estimación de varianza adaptativa y no promediado a cero.

Con respecto al caso de los distintos tamaños de recorte cabe señalar lo siguiente:

Para el caso de tamaños de recorte grandes (1024×1024) se puede concluir que el uso de los distintos parámetros no genera claramente unos resultados mejores con respecto a las otras opciones (la mayor diferencia entre todos los resultados es del 1,67 %). La mejor opción es la que utiliza la varianza adaptativa y no el promediado a cero. Es más, la segunda mejor opción tampoco utiliza el promediado a cero, por lo que se puede concluir que para tamaños de recorte grandes el promediado a cero no aporta mejora alguna, más bien al contrario, empeorando levemente los resultados. Con respecto al tipo de varianza a usar se puede concluir que el uso de la adaptativa requiere mayores tiempos de ejecución. Por tanto, teniendo en cuenta que, en términos de tiempo de ejecución, el uso de la varianza adaptativa es costoso, se puede concluir que para tamaños de recortes grandes y un gran número de imágenes a analizar es preferible no utilizarla (en el peor de los casos empeora un 0,5 % los resultados), a no ser que no haya restricciones de tiempo o se posea una alta capacidad de procesamiento.

Para el caso de tamaños de recorte pequeños (512×512) no existen diferencias significativas con respecto al uso de los distintos parámetros. El peor de los casos es el que utiliza la varianza adaptativa y el promediado a cero, concluyendo que para tamaños de recorte pequeños ésta es una mala opción ya que obtiene peores resultados con respecto a las otras opciones (2,34 % en el mejor de los casos). Con respecto al uso de los distintos tipos de varianza y promediado a cero las conclusiones son las mismas que se dan para el caso de tamaños de recorte grandes.

Con el objetivo de evaluar la escalabilidad de esta técnica se ha repetido un experimento utilizando 14 modelos de 7 fabricantes distintos, lográndose una tasa media de acierto del 87,21 %. Los resultados, en principio, sugieren que el método es aplicable a los conjuntos de imágenes de un gran número de cámaras diferentes y, consecuentemente, el método se aprecia como válido para usos potenciales en el análisis forense de imágenes digitales.

Dependiendo del número y tipo de imágenes que se tengan que analizar y la maximización de la tasa de acierto en función del tiempo del ejecución que se desee, el analista forense tiene la posibilidad de ajustar ciertos parámetros en el algoritmo de identificación de la fuente, permitiéndole obtener unos resultados más cercanos a sus necesidades y restricciones de ejecución.

Capítulo 10

Conclusiones y Trabajo Futuro

En este trabajo se han desarrollado diferentes técnicas de análisis forense para la identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles.

Inicialmente se ha realizado una presentación de conceptos generales relacionados con las técnicas presentadas. Se ha mostrado información detallada acerca de la secuencia de procesamiento de diferentes tipos de dispositivos haciendo énfasis en los elementos que distinguen a las cámaras de dispositivos móviles del resto de dispositivos. Asimismo, se ha analizado la información de los metadatos contenidos en las imágenes digitales, señalando los aspectos más relevantes de los distintos formatos de metadatos. Luego se han comentado las razones que justifican el estudio de este tipo de análisis forense, así como las diversas ramas del mismo. Posteriormente, se ha expuesto una clasificación de las distintas técnicas de análisis forense de identificación de la fuente de adquisición para imágenes digitales.

Seguidamente se han revisado los principales trabajos relacionados sobre las distintas técnicas de identificación de la fuente de adquisición para imágenes digitales con independencia del dispositivo utilizado. La mayoría de ellos son para imágenes generadas con DSCs, no siendo válidas en muchos casos las técnicas presentadas para su aplicación a imágenes de dispositivos móviles. Del estudio de la literatura se han obtenidos los elementos o características de las imágenes que pueden ser potencialmente útiles para la identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles.

A continuación se han detallado las contribuciones de esta Tesis. Así, en primer lugar, se ha descrito la aplicación *Theia*. Esta herramienta permite el tratamiento automático de los distintos conjuntos de metadatos Exif de imágenes digitales tanto individualmente como en grupo. Asimismo, *Theia* integra las dos técnicas de identificación de la fuente de adquisición de imágenes digitales especificados en esta Tesis. También se ha mostrado una comparativa de *Theia* con diversas herramientas con fines similares concluyéndose que, desde el punto de vista del análisis forense, *Theia* es la única que ofrece funcionalidades

avanzadas que permiten trabajar con proyectos con numerosas imágenes. Más aún, ninguna de las herramientas presentadas en la comparación posee tratamiento de metadatos a nivel de conjunto de imágenes ni técnica alguna para la identificación de la fuente de adquisición de imágenes.

En segundo lugar se ha realizado un análisis de los metadatos Exif de un banco de imágenes de dispositivos móviles teniendo en cuenta no sólo los aspectos de obtención de la fuente de adquisición de la imagen. En general, con respecto a los metadatos Exif, siendo conscientes de su facilidad de manipulación, puede concluirse que son de gran utilidad, ya que, como desvelan los distintos análisis de grandes bancos de imágenes realizados con *Theia*, los fabricantes los insertan en el proceso de generación de la imagen. Asimismo, ha podido comprobarse que existen metadatos que pueden aportar información relevante al analista forense como son los datos de geoposicionamiento y la imagen en miniatura. El análisis binario manual realizado con *Theia* ha desvelado la existencia de anomalías en el seguimiento de la especificación Exif. Estas crean graves problemas de interoperabilidad entre las distintas aplicaciones que utilizan los metadatos. Se han clasificado las anomalías encontradas en 10 tipos, siendo desgraciadamente bastante habituales ya que los fabricantes son tendentes a insertar anomalías en los metadatos Exif, razón por la cual las aplicaciones forenses deben tener presente este hecho.

En tercer lugar se han especificado dos técnicas de identificación de la fuente de adquisición de imágenes digitales de dispositivos móviles. En ambas técnicas se han utilizado de forma exitosa clasificadores SVM. Estas técnicas se aplican en lo que se denominan *escenarios cerrados*. En este tipo de escenarios las imágenes, cuya fuente de adquisición hay que determinar, pertenecen a un grupo de dispositivos conocidos a priori. Por tanto, la identificación de la fuente de adquisición de las imágenes está acotada a un número de dispositivos determinado y conocido.

La primera técnica trabaja con un conjunto de 25 características basadas en el ruido del sensor y en la transformada wavelet. Se han realizado diversos experimentos para 2, 5, 7 y 10 dispositivos móviles utilizando distinto número de imágenes en la fase de clasificación, obteniéndose una tasa media de acierto del 94,98 % para el conjunto de los experimentos. La tasa máxima de acierto obtenida en los distintos experimentos ha sido del 96,67 %, habiendo correspondido al caso de 2 dispositivos.

La segunda técnica emplea un conjunto de 81 características basadas en el patrón de no-uniformidad de la foto-respuesta del sensor de ruido. Con respecto a los resultados de los experimentos realizados con esta técnica se concluye que utilizando un tamaño de recorte suficientemente grande (mayor o igual a 1024×1024 aproximadamente) los resultados son buenos, ya que para 6 y 14 dispositivos las tasas mínimas de acierto han sido del 96,33 % y del 87,21 %, respectivamente. La tasa media de acierto obtenida ha sido del 91,99 % para el conjunto de los experimentos. La tasa máxima de acierto obtenida

en los distintos experimentos ha sido del 97,83 %, habiendo correspondido al caso de 6 dispositivos móviles.

Las dos técnicas de identificación presentadas tienen estrecha relación, ya que todas están basadas en el uso de características del contenido de la imagen. *Theia* permite utilizar los diversos conjuntos de características con total versatilidad. Por tanto, el analista forense, para un mismo conjunto de imágenes, puede realizar distintos análisis utilizando diferentes combinaciones de los conjuntos de características. Dependiendo de aspectos como el número o tipo de dispositivos, las características de los mismos, si existen dispositivos de la misma marca, etc., el analista forense utilizará la configuración más adecuada. Todas las técnicas presentadas presentan distintas configuraciones que permiten adaptar el uso de los diferentes conjuntos de características a distintas finalidades.

Por último, conviene subrayar que en todos los experimentos realizados existen otros aspectos relevantes a tener en cuenta como el tamaño del recorte de la imagen y el número de imágenes a utilizar para la fase de entrenamiento del clasificador SVM. Así, puede concluirse que un tamaño de recorte aproximadamente de 1024×1024 píxeles es suficiente para la obtención de buenos resultados, pudiéndose dar el caso de que un recorte mayor empeore los resultados. Con respecto a las imágenes a utilizar en la fase de entrenamiento del clasificador SVM se estima que un número adecuado y suficiente es el de 100 imágenes, tal y como han mostrado los experimentos. Obviamente, en aplicaciones reales, este número dependerá de las imágenes disponibles por el analista forense, pudiéndose utilizar la técnica con otras cantidades de imágenes. Un número menor de imágenes para entrenar el clasificador puede hacer empeorar los resultados de la técnica en cuestión.

10.1 Trabajo Futuro

Como trabajo futuro pueden señalarse las siguientes líneas de investigación, algunas de las cuales ya están en marcha:

- **Identificación de la fuente de adquisición de vídeos digitales** [AGRCSO+14a]: Al igual que las imágenes digitales de dispositivos móviles, los vídeos generados por el mismo tipo de dispositivos son objeto potencial de análisis forense. Son necesarios, por tanto, algoritmos para la identificación de la fuente de adquisición de vídeos digitales. Dado que los vídeos están formados por fotogramas individuales, el objetivo es adaptar los algoritmos de identificación de la fuente propuestos en esta tesis al caso de vídeos digitales. Una propuesta en este sentido se avanza en [AGRCSO+14a].

- **Identificación de la fuente de adquisición de imágenes digitales en escenarios abiertos** [AGRCSO+14b]: Numerosas situaciones de aplicación real de los

algoritmos de identificación de la fuente de adquisición de imágenes digitales pueden darse en los denominados "escenarios abiertos". Es decir, el analista forense no tiene conocimiento a priori del conjunto de dispositivos al que pertenecen las imágenes a clasificar. En estos casos el objetivo no es la identificación de la marca y modelo de las imágenes, sino la agrupación de las imágenes en clases o clusters por modelo de dispositivo fuente. En [AGRCSO+14b] se avanza una propuesta de algoritmo de clustering basado en el uso de las características del patrón del ruido del sensor.

Otras posibles líneas de investigación son:

- **Mejorar las tasas de acierto en la identificación de la fuente de adquisición de imágenes**: Aunque las tasas de acierto de las diferentes técnicas presentadas para la identificación de la fuente de adquisición son altas, existen diferentes aplicaciones a situaciones reales en las cuales se necesitan mejores resultados. Una de esas aplicaciones son las relacionadas con pruebas judiciales donde la tasa de acierto de la técnica debe de ser extremadamente cercana al 100 %. Asimismo, es necesario mejorar las tasas de acierto cuando el número de dispositivos es muy grande, con el objetivo de poder ser utilizado con extensas bases de datos de imágenes de analistas forenses. Para la consecución de este objetivo también es necesario tener en cuenta los aspectos relacionados con la clasificación. En este trabajo para el caso de la identificación de la fuente de adquisición se utilizan máquinas SVM, las cuales se ha demostrado que son útiles para este fin. Sin embargo, se deben investigar nuevos métodos que permitan mejorar las carencias que tienen las máquinas SVM con respecto a la clasificación. Además es necesario mejorar los algoritmos presentados para que sean más robustos con respecto a los distintos posibles ataques. Este aspecto no ha sido tenido en cuenta en la creación de los algoritmos, ya que el principal objetivo ha sido la obtención de buenos resultados para imágenes no manipuladas.
- **Identificación de la fuente de adquisición de imágenes de dispositivos móviles concretos**: Las técnicas de identificación presentadas en este trabajo tienen como objetivo obtener la marca y modelo de la fuente de adquisición de la imagen. Puede darse el caso que este hecho no sea suficiente, ya que no permite diferenciar entre dos dispositivos concretos de la misma marca y modelo.

- **Identificación de la fuente de adquisición de imágenes en cascada**: El número de marcas y modelos de dispositivos móviles es enorme, generándose principalmente dos problemas con respecto a las técnicas de identificación presentadas: bases de datos incompletas y excesivo tiempo de ejecución en grandes bases de datos. Sería conveniente la creación de algoritmos que utilicen las técnicas de identificación de la fuente de adquisición presentadas aplicándolas en cascada, con el objetivo de mejorar los tiempos de ejecución y los resultados de las tasas de identificación (o poder aportar aproximaciones sobre la marca y modelo de una cámara similar si la imagen a identificar no está en la base de datos).

Bibliografía

[Ado92] Adobe Developers Association. TIFF Revision 6.0. http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf, 1992.

[AG11] D. M. Arenas González. Análisis Forense de Imágenes de Móviles Mediante el Uso de Metadatos. Tesis de Máster 13507, Universidad Complutense de Madrid, Noviembre 2011.

[AGRCSO+14a] D. M. Arenas González, J. Rosales Corripio, A. L. Sandoval Orozco, H. J. Romo Torres, and L. J. García Villalba. Identificación de la Fuente en Vídeos de Dispositivos Móviles. In *Actas del XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 265–270, Alicante, España, Septiembre 2014.

[AGRCSO+14b] D. M. Arenas González, J. Rosales Corripio, A. L. Sandoval Orozco, J. A. Zapata Guridi, and L. J. García Villalba. Clasificación sin Supervisión de Imágenes de Dispositivos Móviles. In *Actas del XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 271–276, Alicante, España, Septiembre 2014.

[ALE14] Alexa Top 500 Global Sites. http://www.alexa.com/topsites, 2014.

[AM12] T. Ahonen and A. Moore. Tomi Ahonen Almanac 2012: Mobile Telecoms Industry Annual Review. http://www.tomiahonen.com/ebook/almanac.html, 2012.

[AM14] T. Ahonen and A. Moore. Tomi Ahonen Almanac 2014: Mobile Telecoms Industry Annual Review. http://communities-dominate.blogs.com/, 2014.

[APS98] J. Adams, K. Parulski, and K. Spaulding. Color Processing in Digital Cameras. *IEEE Micro*, 18(6):20–30, December 1998.

[AZ06] M. Al-Zarouni. Mobile Handset Forensic Evidence: a Challenge for Law Enforcement. In *Proceedings of the 4th Australian Digital Forensics Conference*, pages 1–10, Perth Western, Australia, December 2006.

[Bae10] R. Baer. Resolution Limits in Digital Photography: The Looming End of the Pixel Wars. In *Proceedings of the Imaging Systems Conference*, Tucson, Arizona United States, June 2010.

[BL04] M. Boutell and J. Luo. Photo Classification by Integrating Image Content and Camera Metadata. In *Proceedings of the 17th International Conference on Pattern Recognition*, volume 4, pages 901–904, Cambridge, UK, August 2004.

[BL05] M. Boutell and J. Luo. Beyond Pixels: Exploiting Camera Metadata for Photo Classification. *Pattern Recognition*, 38(6):935–946, June 2005.

[BSM06] S. Bayram, H. T. Sencar, and N. Memon. Improvements on Source Camera-Model Identification Based on CFA Interpolation. In *Proceedings of the International Conference on Digital Forensics*, pages 24–27, Orlando, Florida, USA, February 2006.

[BSM08] S. Bayram, H. T. Sencar, and N. Memon. Classification of Digital Camera-Models Based on Demosaicing Artifacts. *Digital Investigation*, 5(1-2):49–59, September 2008.

[CAS+06] O. Celiktutan, I. Avcibas, B. Sankur, N. P. Ayerden, and C. Capar. Source Cell-Phone Identification. In *Proceedings of the IEEE 14th Signal Processing and Communications Applications*, pages 1–3, Antalya, Turkey, April 2006.

[CESR12] F. D. O. Costa, M. Eckmann, W. J. Scheirer, and A. Rocha. Open Set Source Camera Attribution. In *Proceedings of the 25th Conference on Graphics, Patterns and Images*, pages 71–78, Ouro Preto, Brazil, August 2012.

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

[Cho06] K. S. Choi. Source Camera Identification Using Footprints From Lens Aberration. In *Proceedings of the Digital Photography II*, number 852 in 6069, San Jose, CA, USA, February 2006.

[CJJYJwHG07] J. Chul-Jin, L. Ji-Yeon, L. Jeong-won, and C. Hwan-Gue. Smart Management System for Digital Photographs Using Temporal and Spatial Features with EXIF Metadata. In *Proceedings of the 2nd International Conference on Digital Information Management*, volume 1, pages 110–115, Lyon, France, October 2007.

[CK10] H. Cao and A. Kot. Mobile Camera Identification Using Demosaicing Features. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 1683–1686, Paris, France, May 2010.

[CL] C. C. Chang and C. J. Lin. LIBSVM: A Library for Support Vector Machines. http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[Com10] Standardization Committee. Exchangeable Image File for Digital Still Cameras: Exif version 2.3. www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf, 2010.

[Com13] Standarization Committee. Exchangeable Image File for digital still cameras: Exif version 2.3, April 26, 2010. http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf, 2013.

[CSA08] O. Celiktutan, B. Sankur, and I. Avcibas. Blind Identification of Source Cell-Phone Model. *IEEE Transactions on Information Forensics and Security*, 3(3):553–566, September 2008.

[dOCSE+14] F. de O. Costa, E. Silva, M. Eckmann, W. J. Scheirer, and A. Rocha. Open Set Source Camera Attribution and Device Linking. *Pattern Recognition Letters*, 39:92–101, April 2014.

[Exs07] Exsoftware. Photoinfoex. http://www.photoinfoex.com/, 2007.

[Gar14] Gartner Inc. Gartner Says Worldwide Tablet Sales Grew 68 Percent in 2013, With Android Capturing 62 Percent of the Market. http://www.gartner.com/newsroom/id/2692318, 2014.

[GBK+01] Z. J. Geradts, J. Bijhold, M. Kieft, K. Kurosawa, K. Kuroki, and N. Saitoh. Methods for Identification of Images Acquired with Digital Cameras. In *Proceedings of the Enabling Technologies for Law Enforcement and Security*, volume 4232, pages 505–512, Boston, Massachusetts, USA, February 2001.

[GKWB07] T. Gloe, M. Kirchner, A. Winkler, and R. Bohme. Can We Trust Digital Image Forensics? In *Proceedings of the 15th International Conference on Multimedia*, pages 78–86, Augsburg, Germany, September 2007.

[HAJY10] J. S. Ho, O. C. Au, Z. Jiantao, and G. Yuanfang. Inter-channel Demosaicking Traces for Digital Image Forensics. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 1475–1480, Suntec City, Singapore, July 2010.

[Ham] E. Hamilton. JPEG File Interchange Format. Version 1.02, September 1, 1992. http://www.w3.org/Graphics/JPEG/jfif3.pdf.

[Har05] P. Harvey. ExifTool. http://www.sno.phy.queensu.ca/~phil/exiftool/, 2005.

[HCL03] C. W. Hsuand, C. C. Chang, and C. J. Lin. A Practical Guide to Support Vector Classification. Practical Guide, Department of Computer Science and Information Engineering, National Taiwan University, April 2003.

[HLZ10] Y. Hu, C.-T. Li, and C. Zhou. Selecting Forensic Features for Robust Source Camera Identification. In *Proceedings of the International Computer Symposium*, pages 506–511, Tainan, China, December 2010.

[IC 13] IC Insights. Embedded Imaging Takes Off as Stand-alone Digital Cameras Stall. http://www.icinsights.com/news/bulletins/Embedded-Imaging-Takes-Off-As-Standalone-Digital-Cameras-Stall, 2013.

[Inc14] Gartner Inc. Gartner Says Annual Smartphone Sales Surpassed Sales of Feature Phones for the First Time in 2013. http://www.gartner.com/newsroom/id/2665715, 2014.

[ITPI07] The International, Press Telecommunications, Canadian Press, and Adobe Illustrator. International Press Telecommunications Council. *Image Rochester NY*, pages 3–5, 2007.

[JKCS11] F. Jiayuan, A. C. Kot, H. Cao, and F. Sattar. Modeling the EXIF-Image Correlation for Image Manipulation Detection. In *Proceedings of the IEEE International Conference on Image Processing*, pages 1945–1948, Brussels, Belgium, September 2011.

[KMC+06] N. Khannaa, A. K. Mikkilinenib, G. T. Chiub, J.P. Allebacha, and E. J. Delpa. Forensic Classification of Imaging Sensor Types. RFC, Purdue University, February 2006.

[KMD09] N. Khanna, A. K. Mikkilineni, and E. J. Delp. Scanner Identification using Feature-based Processing and Analysis. *IEEE Transactions on Information Forensics and Security*, 4(1):123–139, 2009.

[Kow10] M. Kowalski. ExifPro Image Viewer. http://www.exifpro.com/, 2010.

[LFG06] J. Lukas, J. Fridrich, and M. Goljan. Digital Camera Identification from Sensor Pattern Noise. *IEEE Transactions on Information Forensics and Security*, 1(2):205–214, June 2006.

[LH06] Y. Long and Y. Huang. Image Based Source Camera Identification Using Demosaicking. In *Proceedings of the 8th IEEE Workshop on Multimedia Signal Processing*, pages 419–424, Victoria, British Columbia, Canada, October 2006.

[Li09] C. T. Li. Source Camera Linking Using eEnhanced Sensor Pattern Noise Extracted from Images. In *Proceedings of the 3rd International Conference on Crime Detection and Prevention*, pages 1–6, London, UK, December 2009.

[LLC+12] Q. Liu, X. Li, L. Chen, H. Cho, A. P. Cooper, Z. Chen, M. Qiao, and A. 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*, Dalian, China, June 2012.

[LS11] C. T. Li and R. Satta. On the Location-Dependent Quality of the Sensor Pattern Noise and its Implication in Multimedia Forensics. In *Proceedings of the 4th International Conference on Imaging for Crime Detection and Prevention*, pages 1–6, London, UK, November 2011.

[LTEK07] V. Lanh Tran, S. Emmanuel, and M. S. Kankanhalli. Identifying Source Cell Phone using Chromatic Aberration. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 883–886, Beijing, China, July 2007.

[Lyo01] M. Lyons. EXIF Information Reader. http://www.tawbaware.com/exifread.htm, 2001.

[McK07] C. McKay. Forensic Analysis of Digital Imaging Devices. Technical Report, University of Maryland, 2007.

[MKY08] F. J. Meng, X. W. Kong, and X. G. You. Source Camera Identification Based on Image Bi-Coherence and Wavelet Features. In *Proceedings of the Fourth Annual IFIP International Conference on Digital Forensics*, Kyoto, Japan, January 2008.

[MSGW08] C. Mckay, A. Swaminathan, H. Gou, and M. Wu. Image Acquisition Forensics: Forensic Analysis to Identify Imaging Source. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1657–1660, Las Vegas, Nevada, USA, June 2008.

[Nak05] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, Boca Raton, FL, USA, August 2005.

[OA11] L. Ozparlak and I. Avcibas. Differentiating Between Images Using Wavelet-Based Transforms: A Comparative Study. *IEEE Transactions on Information Forensics and Security*, 6(4):1418–1431, December 2011.

[Oht08] J. Ohta. *Smart CMOS Image Sensors and Applications*. CRC Press, 2008.

[Pla00] J. Platt. AutoAlbum: Clustering Digital Photographs Using Probabilistic Model Merging. In *Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries*, pages 96–100, Hilton Head Island, SC, June 2000.

[Ras07] A. Raskin. Exif Viewer 1.81. https://addons.mozilla.org/es-es/firefox/addon/exif-viewer/, 2007.

[RCAGSO+13] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, L. J. García Villalba, J. C. Hernandez-Castro, and S. J. Gibson. Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform. In *Proceedings of the 5th International Conference on Imaging for Crime Detection and Prevention*, pages 1–6, London, UK, December 2013.

[RCAGSOGV14] J. Rosales Corripio, D. M. Arenas González, A. L. Sandoval Orozco, and L. J. García Villalba. Identificación de la Fuente de Imágenes de Dispositivos Móviles basada en el Ruido del Sensor. In *Actas del XIII Reunión Española sobre Criptología y Seguridad de la Información*, pages 277–280, Alicante, España, Septiembre 2014.

[RCC+08] N. L. Romero, V. G. Chornet, J. S. Cobos, A. S. Carot, F. C. Centellas, and M. C. Mendez. Recovery of Descriptive Information in Images From Digital Libraries by Means of EXIF Metadata. *Library Hi Tech*, 26(2):302–315, 2008.

[RH99] T. Randen and J. H. Husøy. Filtering for Texture Classification: A Comparative Study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4):291–310, April 1999.

[RSBG11] A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein. Vision of the Unseen: Current Trends and Challenges in Digital Image and Video Forensics. *ACM Computing Surveys*, 43(4):1–42, October 2011.

[RSYD05] R. Ramanath, W. E. Snyder, Y. Yoo, and M. S. Drew. Color Image Processing Pipeline. *IEEE Transactions on Signal Processing*, 22(1):34–43, January 2005.

[SOAGGVHC12a] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández-Castro. Análisis Forense de Imágenes de Dispositivos Móviles Utilizando los Metadatos Exif. In *Actas del XXVII Simposium Nacional de la Unión Científica Internacional de Radio*, Elche, Alicante, España, Septiembre 2012.

[SOAGGVHC12b] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández-Castro. Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles. In *Actas del XII Reunión Española sobre Criptología y Seguridad de la Información*, Donostia-San Sebastián, España, Septiembre 2012.

[SOAGGVHC14] A. L. Sandoval Orozco, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández Castro. Analysis of Errors in Exif Metadata on Mobile Devices. *Multimedia Tools and Applications*, 68(1):1–29, January 2014.

[SOAGRC+14] A. L. Sandoval Orozco, D. M. Arenas González, J. Rosales Corripio, L. J. García Villalba, and J. C. Hernández Castro. Source Identification for Mobile Devices, Based on Wavelet Transforms Combined with Sensor Imperfections. *Computing*, 96(9):829–841, September 2014.

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

[SORCAG+13] A. L. Sandoval Orozco, J. Rosales Corripio, D. M. Arenas González, L. J. García Villalba, and J. C. Hernández Castro. Techniques for Source Camera Identification. In *Proceedings of the 6th International Conference on Information Technology*, pages 1–9, Amman, Jordan, May 2013.

[SWL09] A. Swaminathan, M. Wu, and K. J. R. Liu. Component Forensics. *IEEE Transactions on Signal Processing*, 26(2):38–48, 2009.

[Tes05] J. Tesic. Metadata Practices for Consumer Photos. *IEEE Multimedia*, 12(3):86–92, September 2005.

[TLL07] M. J. Tsai, C. L. Lai, and J. Liu. Camera/Mobile Phone Source Identification for Digital Forensics. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pages 221–224, Honolulu, Hawai, USA, April 2007.

[TNC10] V. L. L. Thing, K. Y. Ng, and E. C. Chang. Live Memory Forensics of Mobile Phones. *Digital Investigation*, 7:74–82, August 2010.

[VCEK07] T. Van Lanh, K. S. Chong, S. Emmanuel, and M. S. Kankanhalli. A Survey on Digital Camera Image Forensic Methods. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pages 16–19, Beijing, China, July 2007.

[Wan10] M. Wandel. Exif Jpeg Header Manipulation Tool. http://www.sentex.net/~mwandel/jhead/, 2010.

[WGKM09] B. Wang, Y. Guo, X. Kong, and F. Meng. Source Camera Identification Forensics Based on Wavelet Features. In *Proceedings of the International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 702–705, Kyoto, Japan, September 2009.

[WY06] C. Y. Wen and K. T. Yang. Image Authentication for Digital Image Evidence. *Forensic Science Journal*, 5(1):1–11, September 2006.

[XMP13] XMP Specification: Adobe XML Metadata Framework. http://www.adobe.com/products/xmp, 2013.

Parte II

Resumen de la Investigación en Inglés

Capítulo 11

Image Source Acquisition Identification of Mobile Devices

11.1 Introduction

The current demand for mobile devices (mobile phones, smartphones, tablets, etc.) is increasing year by year despite the global economic crisis. According to Gartner [Inc14] in 2013 smartphone sales grew 42.3 % from the previous year; outnumbering the sales of feature phones for the first time. In total, according to estimates by the ITU, there are 6.8 billion mobile phone subscriptions worldwide, which is a large increase on the 6 billion subscriptions in 2012 and 5.8 billion in 2011.

Having used figures to describe and illustrate the extent mobile devices are used across the world, we must not overlook the importance of having such devices available in our day to day lives. Increasing storage capacity, usability, portability and affordability, have allowed mobile devices to be present in several activities, places and events of daily life. So much so, that according to [AM12], a large number of people have and use more than one mobile device and a typical user turns to their mobile devices an average of 150 times a day.

The majority of these mobile devices have an integrated digital camera, which in contrast to DSC are carried by their owners all the time to most places they attend. According to [IC 13], the share of digital camera sales in mobile phones will be 48 %, with the share DCS sales in this year only 27 %. There are also predictions that the DSCs may disappear in favor of new integrated mobile device cameras [Bae10], because the improved quality of these devices is growing at an unstoppable rate.

Because extensive use of mobile device digital cameras has generated controversy, discussions and rules have been made for the prohibition of using them in places such as government offices, schools and businesses, etc. A consequence of its widespread use, is that digital images can be used as silent witnesses in judicial proceedings (child pornography, industrial espionage, social networks, ...), and in many cases crucial pieces of evidence in a crime [AZ06]. For these reasons, nowadays, digital image forensic analysis of mobile devices is very important.

Forensic analysis of digital images can be mainly divided into two branches [GKWB07]: tamper detection and image source identification. The first of the branches try to discern if an image has suffered any kind of processing after its creation, that is, the image has not been manipulated. The second branch will be presented in this work and it has the aim of identifying the type (camera, scanner, computer) or class (make and model) of the image source acquisition.

11.2 Digital Image Pipeline

To understanding digital images forensic it is essential to know in detail the image acquisition process in digital cameras. This process is summarized in Figure 11.1.

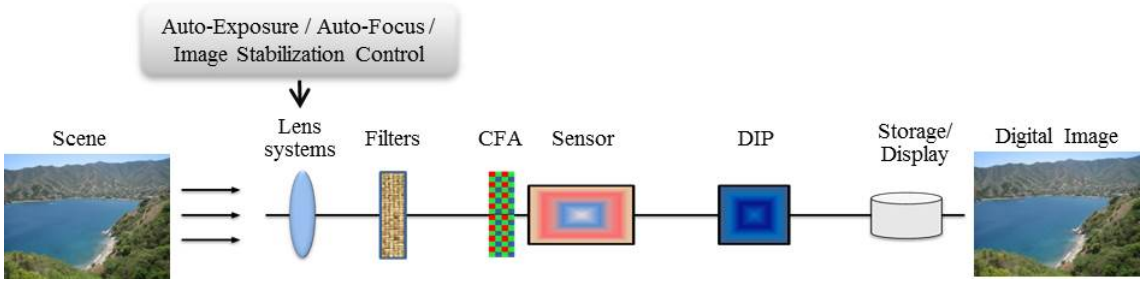


Figura 11.1: Image acquisition process in a digital camera

In order to generate a digital image first, the lens system collects the light from the scene controlling exposure, focus, and image stabilization. Then, the light enters the camera through the lens, it goes through a combination of filters (at least the infra-red and anti-aliasing filters) to ensure maximum image quality. In order to produce a color image the CFA is used. After, light is focused onto the imaging sensor that is an array of light-sensing elements called pixels. After light impacts against pixels they generate an analog signal proportional to the intensity of light, which is converted into a digital signal to be processed by the DIP. Finally, the complete final image is formed by the DIP which performs some operations such as demosaicing, white point correction, gamma correction, compression, etc., aiming to produce a visually pleasing image.

11.3 Image Metadata

Digital images are stored in a variety of formats such as TIFF, JPEG or PSD among others formats. Each image format has different rules regarding how the different metadata formats are stored along with the file itself. Some of the different metadata containers for the diverse formats are: IFDs Exif/TIFF, Adobe XMP and IPTC-IIM. Each one of these containers has its own format indicating the stored metadata properties, as well as their order and codification. In each container there is a separation by semantic criteria.

These semantic groups are divided themselves into individual metadata properties. Each property has some specific data associated such as strings, numbers or arrays. Some properties, like image orientation, are not common to the different standard containers. However, others like copyright strings can be stored in several containers with similar information yet different semantic or slightly different structure (Figure 11.2).

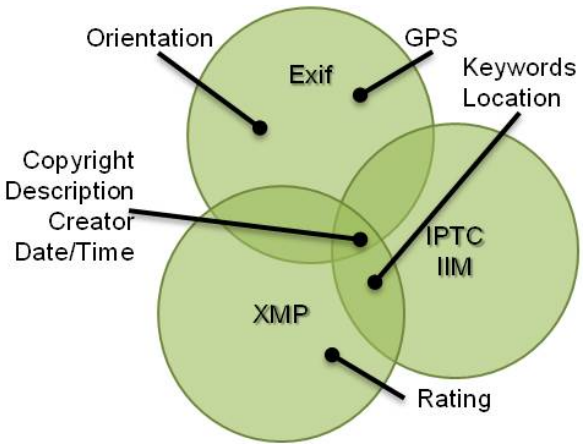


Figura 11.2: Metadata containers

The structural complexity previously described it use to cause problems in the effective and efficient use of the metadata. These problems cause frustration and mistrust in the users about the different metadata systems, because the main aim is looking for the interoperability between the different products and digital image services. Aware of these problems the makers invest a great quantity of resources to solve them. So there are groups of makers such a Metadata Working Group with the objective to relieve or wipe the problems previously described. This group is formed by enterprises such as Apple, Adobe, Canon, Microsoft, Nokia and Sony. Even with the existence of this group the problems are not completely solved, due to the great variety of makers. It is important to highlight that this does not imply the worthlessness of the use of metadata in images. Nowadays we can be sure that the metadata are essential and inseparable in a digital image.

11.4 Forensic Analysis Techniques in Digital Images

In this section we describe the main techniques of digital image forensics for identifying the source of image acquisition and the main work of the analysis. Other compendiums of techniques may be shown in [SWL09] [RSBG11]. The success of these techniques depends on the assumption that all the images acquired by the same device have intrinsic features. The features which are used to identify the make and model of a digital camera are derived from the differences between the techniques of image processing technologies and the components which are used. The biggest problem with this approach is that different models of digital cameras use components of a small number of manufacturers, and the algorithms used are also very similar between models of the same brand. According to [VCEK07] for this purpose four groups of techniques can be established depending on their base: lens system aberrations, CFA interpolation, image characteristics, and sensor imperfections. In addition to the above there is another group of techniques based on metadata.

11.4.1 Techniques Based on Metadata

Digital cameras have a powerful source of information which is the embedded metadata in digital images files. Metadata, or "data about data" store information related to the conditions of image capture, as the date and time of acquisition, flash presence or absence, object distance, exposure time, shutter opening and GPS information among others. In other words, it provides relevant information to supplement the main content of a digital document.

The Exif specification [Com13] is the most common container of metadata in digital cameras [Bae10]. The Exif specification includes hundreds of labels, among which are *make* and *model*, although it should be noted that the specification does not make their existence in the image files compulsory.

Techniques based on the image metadata are the simplest. There are plenty of studies focused on the different types of metadata, both for finding information and for image classification [Pla00] [BL05] [Tes05] [RCC+08] [AG11]. Metadata can also be used as input or aid for other forensic techniques. For instance, in the application of content-based image techniques, Exif metadata can provide a large number and variety of technical information, which may allow an increase in the success rates or improve the results of the application of certain forensic algorithms [BL04] [JKCS11] [CJJYJwHG07].

However, these techniques depend largely in the metadata inserted by manufacturers when the image is created and the correction. Moreover, this method is the most vulnerable to malicious alterations.

11.4.2 Techniques Based on Lens Aberration

During the image generation process the lens system can introduce some aberrations. There are several types of aberrations: spherical, coma, astigmatism, field curvature, radial distortion and chromatic aberration. The radial distortion is the one with the most impact over pictures, especially in cameras with cheap wide angle lenses. Most digital cameras use this type of lens for economic reasons.

In [Cho06] the lens radial distortion is proposed as the best technique for source identification. The authors conclude that each camera model expresses a unique pattern of radial distortion which helps to uniquely identify it. They experimented with three different cameras and obtained accuracy between 87 % and 91 % in identifying the source camera.

[LTEK07] proposes lateral chromatic aberration as a technique for identifying source camera. The authors performed experiments using little sets of cameras with non-modified images or modified images with random crops regions. In the experiment in which three cameras of different brands were used 86.67 % accuracy in identifying the source was obtained. It was also concluded that this technique is not suitable for identifying the source of different camera models from the same brand.

11.4.3 Techniques Based on CFA Interpolation

Some authors consider that the choice of the CFA matrix and the specification of color interpolation algorithms produce some of the most significant differences between different camera models [BSM06] [CAS+06] [LH06] [BSM08].

Commercial cameras have a single sensor instead of multiple sensors for each component color. In essence, the color interpolation introduces a specific type of correlation between the color values of image pixels. The specific form of these dependencies can be extracted from the images to differentiate the color interpolation algorithms and determine make and model of the camera that generated an image.

In [LH06] use correlations between pixels for the source identification. Neutral networks are used for classification. The method was tested with cartoon images from four cameras and the success rate obtained was between 95 % and 100 %, with an average accuracy of 98.25 %. Tests for modified images were also made with 80 % success rate for a 80 % JPEG compression. Since the cameras from the same manufacturer use the same color interpolation algorithm, this approach is not efficient at differentiating between different models from the same maker. Also, as shown in the experiments, good results are not obtained when the images have been modified or when they have high compression level.

In [CAS+06] a set of binary similarity measures is used as metrics to estimate the similarity between image bit planes. This work uses a set of 108 binary similarity measures. The success rate of their experiments was between 81 % and 98 % to classify three cameras and decreased to 62 % to identify between nine cameras. It can clearly be seen that the results of the method depends on the number of cameras used in the experiments.

In [BSM08] an algorithm for identifying and classifying color interpolation operations is presented. This proposal is based on two methods to perform the classification process: first using an algorithm to analyze the correlation of each pixel value with its neighbors' values, and secondly an analysis of the differences between pixels independently. Different experiments with different numbers of cameras and image types were performed. The accuracy for the source camera identification had between 84.8 % and 92.56 % of average success rate.

In [CK10] a technique for source identification based on the information of the CFA matrix interpolation process and a comparison with other techniques is presented. This technique has three new sets of demosaicing features: weights, EC and NGS. Since the number of features is very high a process (ERE) is performed to decrease the number of it. Different experiments were performed using classifiers 1NN and probabilistic support vector machine PSVM. The results using 15 cameras from four different manufacturers and 11 different models (there are cameras of the same brand and model), with a reduction to 20 features and PSVM classifier, obtained an average success rate of 99.4 % for the brand identification and 94.8 % for model identification.

[HAJY10] proposes four algorithms which are based on aspects of inter-channel correlation. These algorithms calculate variance maps (v-maps) and classify using 1NN. The experiments image source identification uses four cameras for three different manufacturers and 50 images of each camera (25 for training and 25 test). The results show an average accuracy of 94.5 % and the authors conclude that the inter-channel correlation provides a complementary approach to previous studies which dealing with correlations between pixels introduced in the demosaicing process.

11.4.4 Techniques Based on Image Features

These techniques use a set of features extracted from the content of the image to identify the source. These features are divided into three groups: color features, IQM and wavelet domain statistics.

[TLL07] proposes a method to identify the source using the following features: color features, image quality metrics and frequency domain. The study adopted the wavelet transforms as a method to calculate the wavelet domain statistics and use a SVM for

classification. In experiments digital cameras and mobile devices were used. The results obtained in different experiments show results between 61.7 % and 99.72 % accuracy.

In [MSGW08] authors extend the source identification to different devices such as mobiles, phones, digital cameras, scanners and computers. In this proposal they base it on the differences in the image acquisition process to create two features groups: color interpolation coefficients and noise features. In the experiments they use five smartphone models, five digital camera models and four scanner models to identify the source type. Their experiments showed an overall result of 93.75 % accuracy. Identifying the maker and model of five mobile phone models resulted in an accuracy of 97.7 %.

In [WGKM09] a method for source camera identification is proposed through the extraction and classification of wavelet statistical features. Finally 216 first-order wavelet features and 135 second order co-occurrence features is obtained. The most representative features are selected using an SFFS algorithm and they are classified using a SVM. Identification success average of 98 % the set of all cameras and an average success rate of 96.9 % for the three cameras of the same model is achieved.

[HLZ10] performs experiments with common imaging features to identify the source: wavelet, color, IQM, statistical features of difference images and statistical features of prediction errors. In the experiments, different combinations of different types of features are used and a SVM for classification of different devices. Ten different cameras from four different makers with 300 images from each camera (150 for training and 150 for testing) and a resolution of 1024x1024 is used. Using all the features a score of 92 % success rate is obtained. Moreover experiments were performed to check the robustness against three of the most common alterations in digital images: JPEG compression, cropping and scaling. The final conclusions of this work are that some of the feature sets provide good success rates for intact images, but not for images with modifications. It also shows that different types of manipulations have different effects on success rates of different feature sets.

In [OA11] a technique for image source identification is proposed using ridgelets and contourlets subbands statistical models. After the feature extraction a SFFS algorithm is used for feature election and a SVM for classification. The method based on 216 wavelet features is considered useful only for the representation of a dimension, the approach based on ridgelets uses 48 features, and the approach based on contourlets includes a total of 768 features. In experiments with three cameras from different makers success rates are between 99.5 % and 99.8 %. The contourlets and ridgelets are not only effective in differentiating between camera models, but also to differentiate between natural images or those produced by computer, or to differentiate between images from scanners of the same maker. However the authors believe that improvements could be implemented experimenting with different selection algorithms.

In [LLC+12] a method using the marginal density DCT coefficients in low-frequency coordinates and neighboring joint density features from the DCT domain is proposed. Furthermore, hierarchical clustering and SVM is used to detect the source of acquisition of the images. In experiments with images from five smartphone models of four makers an accuracy of between 86.36 % and 99.91 % was obtained, achieving the best results with a linear SVM kernel.

11.4.5 Techniques Based on Using Sensor Imperfections

These techniques are based on the study of fingerprints which can leave sensor defects on pictures. These techniques are divided into two branches: pixel defects and SPN. In the first pixel defects, hot pixels, dead pixels, row or column defects and group defects are studied. In the second pattern noise by averaging multiple noise residuals obtained by any noise removal filter is constructed. The presence of the pattern is determined using a classification method as correlation or SVM.

In [GBK+01] pixel defects of CCD, sensors are studied, focusing on different features to analyze images and then identify their source: CCD sensor defects, the file format used, noise introduced in the image and watermarking introduced by makers. Among the CCD sensor defects are considered hot spots, dead pixels, group defects, and row or column defects. Results indicate that each camera has a different defect pattern. Nevertheless, it is also noted that the number of pixel defects for images from the same camera is different and varies greatly depending in the image content. Likewise, it was revealed that the number of defects varies with temperature. Finally, the study found that high quality CCD cameras do not have this kind of problem. When considering only defective CCD sensors this study is not applicable to the analysis of images generated by mobile devices.

[LFG06] analyzes the sensor pattern noise from a set of cameras, which functions as a fingerprint allowing the unique identification of each camera. This pattern noise is obtained averaging the sensor noise extracted from different images with a noise removal filter. To identify the camera from a given image, the reference pattern is considered as a watermark in the image and its presence is established by a correlation detector. The study was done with approximately 320 images from 9 cameras (2 are exactly the same model) and good results were obtained. It is noted that this success rate is because in the experiments the authors used the same set of images to calculate the reference pattern and correlations. It is also shown that this method is affected by processing algorithms such as image JPEG compression and gamma correction. According to [VCEK07] the results for pictures with different sizes were unsatisfactory. Also in this technique, images whose reference pattern is extracted must have the same size as the test images.

In [CESR12] an approach to source camera identification in open set scenarios is proposed, where unlike closed scenarios it is not assumed to have access to all the possible image source cameras. This proposal includes three phases: definition of regions of interest, determining the characteristics and source camera identification. Different regions of the images can contain different information about the fingerprint of the source camera. Besides, this approach in contrast to others considers 9 different ROIs, not only the central region of the image. Using these ROIs it is possible to work with different resolution. For determining the features the SPN for each of the R, G, B and Y (luminance) channels is calculated, generating a total of 36 representative features for each image. Then, the features of images taken by the camera under investigation are labeled as positive class and features from images made by other cameras as negative classes. After the SVM training phase in which the hyper-plane that separates the positive and negative classes is estimated. Later, the unknown classes of open stage are taking into account, moving the generated hyper-plane toward the positive classes or to the negative classes. By moving the hyper-plane the margin can change to determine if an image belongs to one class or another. This process is called modeling decision boundaries. In the experiments a set of 25 digital cameras of 9 manufacturers, 150 images of each camera in JPEG format with different light configuration, zoom and flash are used. The results of the experiments showed a success rate of 94.49 %, of 96.77 % and 98.10 %, using open sets with 2/25, 5/25 and 15/25 cameras, respectively, defining open x/y as the set of cameras and where x cameras are known and used for training and $y - x$ are unknown cameras, whose images and the images of the known cameras are used in test stage. [dOCSE+14] is an extension of the article, where in addition to presenting other techniques and algorithms, new experiments are performed. In experiments 13210 images of 400 cameras were used (they only have physical access to 25 cameras, the rest are images downloaded from *Flickr*) and the best results obtained success rates of 96.56 %, 97.34 %, 96.80 % and 97.18 %, using open sets with 2/25, 5/25, 10 /25 and 15/25 cameras respectively.

11.5 Tool for Forensic Analysis of Mobile Devices Pictures

It is obvious that the metadata retrieval is tedious and slow. Therefore, tools are needed for automatic extraction and graphical visualization in a user friendly way.

Theia, the developed application, carries out two levels of picture analysis: individual and in groups. The former obtains the Exif information of a single picture, finds alterations after comparing it to the thumbnail and automatically places it in Google Maps and Google Earth. The different analysis which can be executed over each group are: picture administration (adding or removing pictures), preset queries, modification analysis based on the stored thumbnail, advanced queries and image geopositioning.

After analyzing different images there are several cases detected in which the specification is not completely followed, even though the heading indicates the opposite. To identify errors *Theia* analyzes each Exif metadata tags using the standard Exif 2.3. The results of these experiments allow us to conclude that many of the manufacturers do not follow the specification Exif to 100 % indicating in the file itself otherwise. This can cause serious problems in extracting metadata from images through applications and interoperability issues between different devices.

The experiments about Exif errors were carried out using a database of 4000 photos of different models of mobile phones of several of the best-known brands. First, we made an analysis by brand, and then by the most problematic devices, in order to focus on the most common errors and finally inspect which are the most error prone tags. *Theia* has made a classification of 10 types of errors in Exif metadata showing examples of these anomalies.

It is remarkable that brands like BlackBerry (99.79 %), HP (100 %) and Palm (100 %) have a high error rate in implementing the Exif 2.3 specification. Samsung and LG also have a high rate of photos with errors (87.97 % and 89,62 % respectively). Apple has a big percentage of erroneous photos due to errors of type 9 as their photos with GPS information do not follow the specification. However, Sony Ericsson was the best performing obtaining a 4.69 % of erroneous photos, which is far lower than the next lowest error rate brand, Nokia with 54.40 % percent of erroneous photos. Figure 11.3 shows the percent of the photos with errors in their Exif metadata from the included brands in the database. We have also made a lot of different experiments.

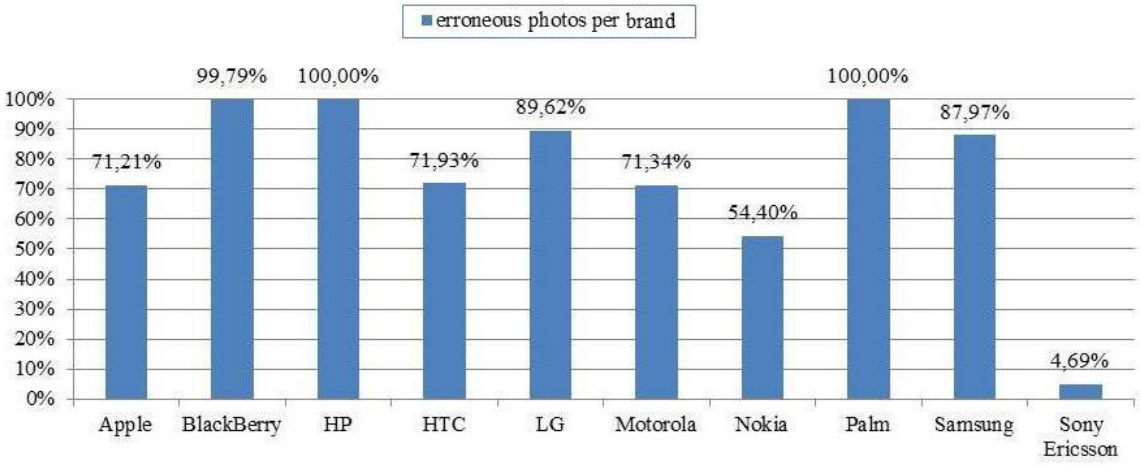


Figura 11.3: Percent of erroneous photos per brand

The metadata extraction applications that exist in the market, such as *PhotoInfoEx* [Exs07], *Exif Viewer* [Ras07], *EXIFRead* [Lyo01], *ExifTool* [Har05] y *Jhead* [Wan10], do not take into account the management of this kind of errors, mistakenly assuming that the manufacturers follow strictly the Exif specification. This may mean that the extracted information is not right, or that it does not extract potentially valid information for the forensic analyst due to the existence of an error in the following of the Exif specification by manufacturers.

Theia compensates for these shortcomings and in some cases it offers the analyst alternatives in the obtaining of the information when the following of Exif specification is not correct. That is to say, *Theia* not only detects the 10 defined errors, but in certain kinds of errors, it provides options for the extraction of data which are potentially valid from the forensic viewpoint.

11.6 Wavelet Transform and Sensor Imperfections

In this section, we describe how features extracted can be utilized to more reliably detect the individual source camera of an image. Here the two types of methods (sensor imperfections and wavelet transforms) are combined to more accurately detect source camera of an image.

During the image generation process usually several defects are injected; these will appear in the final image as noise. One type of noise is caused by array's defects; this includes hot point defects, dead pixels, pixel traps, cluster defects, and column defects. This causes those pixels to differ greatly from the original image, in several cases being indifferent to which of the two images is taken, since they all the time show the same pixel value. For instance, the dead pixels will appear in the image as a black pixel in the resulting image, and hot pixels will appear as brilliant points.

There are some filters to smooth out the effect of this noise. For simplicity, speed and ease of implementation we will use the Gaussian filter. This filter will be used later to eliminate quickly and effectively the noise in the images, and with these data we will be able to perform different operations that lead us to determine the different features.

Our aim is to obtain a set of image features that enable us to clearly differentiate the camera model. Given an initial image of $M * N$ pixels, whith M rows and N columns, we denote I_{noise} as the noise in the original image and $I_{denoised}$ as the image with no noise. Then we get:

$$I_{noise} = I - I_{denoised} \tag{11.1}$$

In order to achieve noise-free images we will use the Gaussian filter, which provides us the necessary speed for analyzing large numbers of photos in a short time. Next, we will subtract each color component (RGB) to the original picture, which will give the noise component of each pixel for each color.

The noise in the original image I_{noise} can be modelled as the sum of two components, the constant noise $I_{noiseconstant}$ and random noise $I_{noiserandom}$.

$$\hat{I}_{noiseconstant}(1,j) = \frac{\sum_{i=1}^{M} I_{noise}(i,j)}{M}, 1 \leq j \leq N \tag{11.2}$$

To identify the similarities between different rows regarding the reference pattern, we will use the correlation of these with the mentioned pattern.

$$correlation(X,Y) = \frac{(X - \overline{X}) \cdot (Y - \overline{Y})}{\|X - \overline{X}\| \cdot \|Y - \overline{Y}\|} \tag{11.3}$$

We do the same for columns. Once we obtain the row and column correlations, we will obtain the features. At the time of obtaining the features it is important to consider that the input photo orientation is critical, as this might change completely the resulting features. We obtain 25 features (16 noise features and 9 wavelet transforms features).

- **First-Order and Higher-Order Features**: For every type of correlation (rows or columns) we obtain the first-order statistics: mean, median, minimum and maximum. The higher-order features are: variance, skewness and kurtosis. Additionally, we add the ratio between the correlations of rows and columns. It was considered appropriate adding a new feature based on the image noise to the set of features above. This new feature measures the medium noise per pixel, which is independent from the columns and rows correlations of the reference pattern. In total we have 7 rows features, 7 columns features, the ratio and average pixel noise, resulting in a total of 16 features.
- **Wavelet Transforms**: Each color band is split into three sub-bands using QMFs and subsequently the mean of each of the three sub bands, giving us a total of 9 features.

Classification was performed using a SVM of the *Radial Basis Function* (RBF) kernel. We used the LibSVM package in which the SVM is extended to multiple classes yielding

class probability estimates [CL]. The kernel parameter $\gamma = 2^3$ and cost parameter $C = 32768$ were used for the SVM. A grid search was used to obtain the best kernel parameters (γ and C). The classifier was trained and tested with feature vectors extracted from randomly selected images.

11.6.1 Results

To verify the effectiveness of the extraction features algorithm for identifying different sources, several experiments were made varying the mobile phone models and the number of pictures, as we will verify how this affects the algorithm. In our first experiment, we will take 10 different phone models and various brands which are listed below: Iphone 3G, Iphone 3GS, Blackberry 8520, HTC Desire HD, LG Ku990i, Nokia 5300, Nokia 6110, Nokia N95, Nokia E61i, Sony and Ericsson W580i.

Of all these models we take exactly 50 photographs. The results show a 89.4558 % accuracy. With so few photographs of each group, the result seems quite remarkable.

We took another group of phones, from which there will be at least 150 photos with a maximum of 200. The 7 considered models are: Blackberry 8520, HTC Desire HD, LG Ku990i, Nokia 5300, Nokia 6110, Nokia 6300, Sony Ericsson T707, and Sony Ericsson W580i. The result in this case is a 94.2227 % accuracy. Therefore, we can observe that the performance of the algorithm is much better as a consequence of having more images, but is also positively affected because we have 3 models less.

With the propose of making a more direct comparison of how the number of classes affects performance, we will compare two common models, 5 groups in total with 50 pictures each, and then we will try with 150 photos each. The common models of each brand are: Blackberry 8520, HTC Desire HD,LG Ku990i, Nokia 5300, and Sony Ericsson W580i.For 50 photos we have an accuracy of 95.8621 %. With 100 images each we have a rate of 96.2 %. We can see that the difference is not huge. We tested with only two groups to verify if this difference could be even smaller, and so we used the Blackberry 8520 and Sony Ericsson W580i.

Table 11.1 shows that the accuracy does not change significantly, so we can conclude that the number of images does not affect significantly the success rate in this approach.

Tabla 11.1: Accuracy rate by number of photos

Number of Photos	Success Rate
30	96.6667 %
60	95 %
90	94.4444 %
120	96.6667 %
150	96.3333 %

We show that the combined set of features can provide tell-tale clues and accurately help trace the origin of the input image and help identify the mobile phone camera brand and model that was used in its capture with high accuracy.

11.7 Sensor Pattern Noise and Wavelet Transform

Previous work has shown sensor pattern noise [GBK+01] [Li09] [LFG06] and wavelet transform [MKY08, OA11] to be an effective method for source camera identification. However, almost all studies have focused only traditional cameras excluding mobile cameras. This makes it an area of study that requires attention. Using a biometric analogy, we consider each noise pattern to be a fingerprint of its source camera's sensor. In our study, sensor pattern noise is used to classify images captured by, camera enabled, smartphones.

The noise pattern of an image refers to any spatial pattern that does not change from one image to another. It is composed of the spatial noise which is independent of the signal (FPN) and for the spatial noise due to the difference in the response of each pixel to the incident signal (PRNU). The noise pattern structure is shown in Figure 11.4.

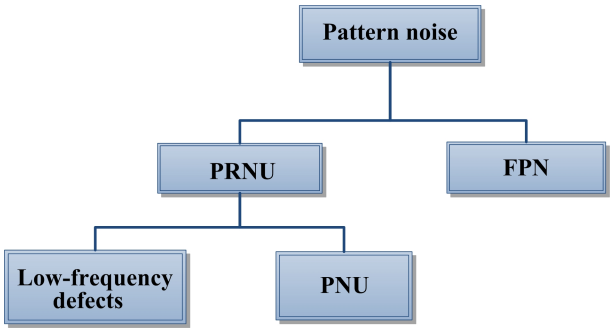


Figura 11.4: Sensor noise pattern

Noise FPN is generated by the dark current and it also depends on exposure and temperature. Since the fixed noise pattern is an independent additive noise, some cameras automatically removed by subtracting a dark frame to generated images.

Noise PRNU is the dominant part of the sensor noise pattern of an image and it is a multiplicative noise dependent. Noise PRNU is mainly formed by the uniformity of pixel PNU and by the low frequency defects as *zoom* settings and light refraction in the dust particles and lenses. Noise PNU is the light sensitivity difference between pixels of the sensor array. It is generated by the lack of homogeneity of the silicon wafers and by the imperfections during the sensor manufacturing process. Due to the nature and origin, it is very unlikely that even the sensors from the same wafer have PNU correlated patterns. This noise is not affected by ambient temperature nor by humidity. Noise PNU is usually more common, complex and significant in CMOS sensors, due to the complexity of pixel array circuitry.

Our approach characterises the fingerprints using wavelet based feature vectors. The scheme presented in Figure 11.5 shows the functional diagram of our proposal.

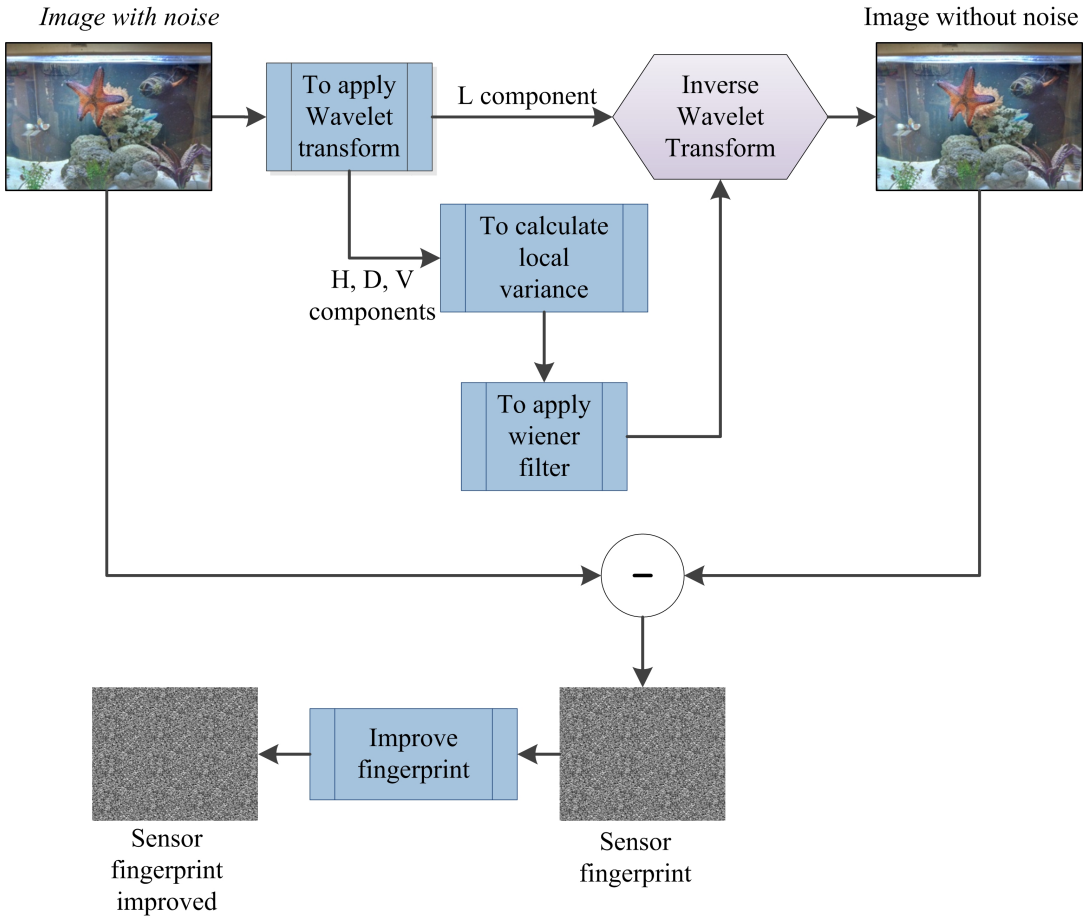


Figura 11.5: Functional scheme

Noise images were obtained using the method previously described by [LFG06] also summarised by Algorithm 3 as follows.

Algoritmo 3: Extracting PRNU

Input: Image I
Variance estimation: adaptive or non-adaptive
Result: Sensor fingerprint I_{noise}

procedure ExtractPRNU(I)
  Apply a wavelet decomposition in 4 levels to I;
  foreach *wavelet decomposition level* **do**
    foreach *component* $c \in \{H, V, D\}$ **do**
      Compute the local variance;
      if *adaptive variance* **then**
        Compute 4 variances with windows of size: 3, 5, 7 and 9;
        Select the minimum variance;
      else
        Compute the variance with a window of size 3;
    Compute noiseless wavelet components applying the Wiener filter to the variance;
  Obtain I_{clean} by applying the inverse wavelet transform with clean components calculated;
  Obtain the sensor noise with $I_{noise}=I-I_{clean}$;
  Apply zero-meaning to I_{noise};
  Increase the green channel weight with $I_{noise}= \quad 0.3{\cdot}I_{noise_R}+0.6{\cdot}I_{noise_G}+0.1{\cdot}I_{noise_B}$;
end procedure

To extract its noise pattern, an image is decomposed into its red, green and blue color channels. Then, a four-level wavelet decomposition of each color channel is calculated using the Daubechies, 8-tap, QMF. The number of decomposition levels can be increased to improve accuracy or reduced to reduce processing time.

Horizonal H, vertical V and diagonal D high-frequency images are obtained for each level of decomposition. For each detail image, the local scene variance in a $W \times W$ window is estimated. Four estimates are obtained with window sizes corresponding to $W \in \{3, 5, 7, 9\}$. Finally, we choose the estimate which maximises the a-posteriori probability.

$$\hat{\sigma}^2(i,j)=\text{máx}\left(0, \frac{1}{W^2}\sum_{(i,j)\in N} c^2(i,j) - \sigma_0^2\right), \quad (i,j)\in J \tag{11.4}$$

Where, $c(i, j)$ is the high-frequency component and $c \in \{H, V, D\}$; σ_0 controls the degree of noise suppression.

The minimum of four variances is chosen as the best estimate:

$$\hat{\sigma}^2(i,j)=\text{mín}\left(\sigma_3^2(i,j),\sigma_5^2(i,j),\sigma_7^2(i,j),\sigma_9^2(i,j)\right), \quad (i,j)\in J \tag{11.5}$$

An alternative, and less accurate method, is to simply use $W = 3$ as the estimated local variance.

The denoised wavelet coefficients are defined by the Wiener filter as follows:

$$c_{clean}(i,j) = \quad c(i,j) \quad \frac{\hat{\sigma}^2(i,j)}{\hat{\sigma}^2(i,j) + \sigma_0^2} \tag{11.6}$$

The noise residual is obtained by calculating the inverse transform and subtracting the denoised image from the original image. JPEG and demosaicing artefacts, present in the noise image, are suppressed by subtracting the mean column and row values [CFGL08]. Greater weight is given to the green channel since due to the configuration of the color matrix this channel contains more information about the image [CSA08] [McK07] [APS98].

The next step is to obtain features that characterise the sensor fingerprint for the purpose of classification. A total of 81 features (3 channels×3 wavelet components×9 central moments) is extracted using the Algorithm 4 as follows:

Algoritmo 4: Extracting features

Input: Sensor fingerprint I_{noise}
Result: 81 features

1. **procedure** ExtractFeatures(I)
2. Separate R, G and B color channels of I_{noise};
3. **foreach** *color channel* **do**
4. Apply a wavelet decomposition in 1 level;
5. **foreach** *component* $c \in \{H, V, D\}$ **do**
6. Compute k central moments with $m_k = \frac{1}{n}\sum_{i=1}^{n} |c_i - \overline{c}|^k$;
7. **end procedure**

11.7.1 Results

To assess the effectiveness of the proposed algorithms, two experiments were conducted considering the central 1024 × 1024 pixel image block, as recommended in [LS11]. Table 11.2 summarises the experimental conditions used in our algorithms.

Tabla 11.2: Parameters used in the proposed algorithms

Parameter	Value
Dimensions	1024 × 1024
Number of training photos by camera	100
Number of testing photos by camera	100
Variance estimation	Non-adaptive

The mobile device digital cameras used and their configurations are showed in Table 11.3.

Tabla 11.3: Configurations used in mobile device digital cameras

Brand	Model	Resolution	Taking Conditions
Apple	iPhone3G (A1)	2 MP (1600 × 1200)	Scene type: Any Orientation: Vertical Flash: Disabled Light: Natural White balance: Auto Digital zoom ratio: 0 Exposure time: 0 seg ISO speed: Automatic
	iPhone4S (A2)	8 MP (3264 × 2448)	
	iPhone3 (A3)	2 MP (1600 × 1200)	
	iPhone5 (A4)	8 MP (3264 × 2448)	
Black Berry	8520 (B1)	2 MP (1600 × 1200)	
Sony Ericsson	UST25a (SE1)	5 MP (2592 × 1944)	
	U5I (SE2)	8 MP (3264 × 2448)	
Samsung	GT-I9100 (S1)	8 MP (3264 × 2448)	
	GT-S5830 (S2)	5 MP (2592 × 1944)	
	GT-S5830M (S2)	5 MP (2592 × 1944)	
LG	E400 (L1)	3.2 MP (2048 × 1536)	
HTC	DesireHD (H1)	8 MP (3264 × 2448)	
Nokia	E61I (N1)	2 MP (1600 × 1200)	

In the first experiment, a group of 8 mobile device digital cameras from 4 different manufacturers was tested. From Apple, the models iPhone3G (A1), iPhone4S (A2), and iPhone3 (A3) were considered; from BlackBerry the 8520 (B1); from Sony Ericsson the UST25a (SE1) and the U5I (SE2); and from Samsung the GTI9100 (S1) and the GTS5830 (S2) models.

Classification was performed using a SVM with the configuration showed in Section 11.6. The performance of the classifier was tested 10 times, using a 10 different random samples of 100 images, and the average classification rate recorded. The performance changed only slightly in each run which indicates stability over different training and testing image sets.

Sample confusion tables from eight camera groups are given below. The best, middle, worst case tables are show in Tables 11.4, 11.5 and 11.6 respectively. The average accuracy for correctly identifying camera make and model was 93.2 %.

Tabla 11.4: Confusion matrix of best result (93.87 %)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**96**	1	0	0	0	0	0	3
A2	0	**97**	0	0	0	0	3	0
A3	0	0	**98**	0	0	0	2	0
B1	0	0	0	**94**	0	4	0	2
SE1	11	1	0	0	**88**	0	0	0
SE2	3	0	0	1	0	**93**	1	2
S1	4	8	0	0	0	3	**85**	0
S2	0	0	0	0	0	0	0	**100**

Tabla 11.5: Confusion matrix of middle result (93.25 %)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**94**	1	0	0	0	1	0	4
A2	0	**96**	0	0	1	0	3	0
A3	0	0	**97**	0	0	0	2	1
B1	0	0	0	**94**	0	2	0	4
SE1	10	1	0	0	**89**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	5	6	0	0	0	6	**83**	0
S2	0	0	0	0	0	1	0	**99**

Tabla 11.6: Confusion matrix of worst result (92.62 %)

Camera	A1	A2	A3	B1	SE1	SE2	S1	S2
A1	**92**	1	0	0	0	0	0	7
A2	0	**96**	0	0	1	0	3	0
A3	0	1	**99**	0	0	0	0	0
B1	0	0	3	**91**	0	4	0	2
SE1	7	2	0	0	**91**	0	0	0
SE2	2	0	0	1	0	**94**	1	2
S1	4	10	0	0	0	7	**79**	0
S2	0	0	0	0	0	1	0	**99**

In the second experiment, in order to evaluate the scalability of the method to a larger number of classes, a group of 14 mobile device digital cameras from 7 different manufacturers was used. From Apple, the models iPhone3G (A1), iPhone4S (A2), iPhone3 (A3) and iPhone5 (A4) were considered; from BlackBerry the 8520 (B1); from Sony Ericsson the UST25a (SE1) and the U5I (SE2); from Samsung the GTI9100 (S1), the GTS5830 (S2) and the GT-S5830M (S3); from Lg the E400 (L1); from HTC the DesireHD (H1) and the Desire (H2); finally from Nokia the E61I (N1) model. The average classification rate dropped to 87.214 % as shown in the confusion matrix of Table 11.7.

Tabla 11.7: Confusion matrix of experiment 2

Camera	A1	A2	A3	A4	B1	SE1	SE1	S1	S1	S3	L1	H1	H2	N1
A1	**90**	0	0	2	0	0	0	0	7	0	1	0	0	0
A2	0	**91**	0	3	0	0	0	3	0	0	0	1	2	0
A3	0	0	**98**	0	0	0	0	2	0	0	0	0	0	0
A4	0	0	1	**88**	0	0	0	0	0	0	3	6	0	2
B1	0	0	0	2	**73**	0	0	0	4	0	0	1	0	20
SE1	7	0	0	0	0	**80**	0	0	0	0	1	12	0	0
SE2	1	0	0	2	2	0	**86**	1	2	5	1	0	0	0
S1	4	5	0	4	0	0	1	**83**	0	0	1	0	2	0
S2	0	0	0	0	0	0	0	0	**100**	0	0	0	0	0
S3	0	0	1	0	0	0	8	0	0	**85**	0	1	0	5
L1	0	0	0	9	0	6	0	0	2	0	**70**	13	0	0
H1	2	0	0	0	0	11	0	0	1	0	1	**85**	0	0
H2	0	6	0	0	0	0	0	0	0	0	0	0	**94**	0
N1	0	0	0	0	2	0	0	0	0	0	0	0	0	**98**

This method for source camera identification, based wavelet features of image noise residuals and SVM classification, was tested on photographs acquired from a range of smartphones. In the first experiment 8 models from 4 manufacturers were considered resulting in an overall accuracy of 93.2 %. In order to evaluate the scalability of the approach, we repeated the experiment using 14 models from 7 manufactures and achieved an average success rate of 87.214 %. Our results, tentatively, suggest that the method is applicable to data sets containing images from a large number of different cameras and therefore the method promises potential utility for digital forensics and data mining applications.

Parte III

Anexo

Apéndice A

Especificación de Theia

A continuación se realiza una presentación pormenorizada de la aplicación desarrollada. Como se ha comentado en el Capítulo 6, *Theia* se divide en dos grandes partes: tratamiento de imágenes a nivel individual y tratamiento de imágenes a nivel de grupo.

A.1 Tratamiento de Imágenes a Nivel Individual

Esta funcionalidad está asociada a la pestaña Exif Info y su apariencia gráfica general puede verse en la Figura A.1.

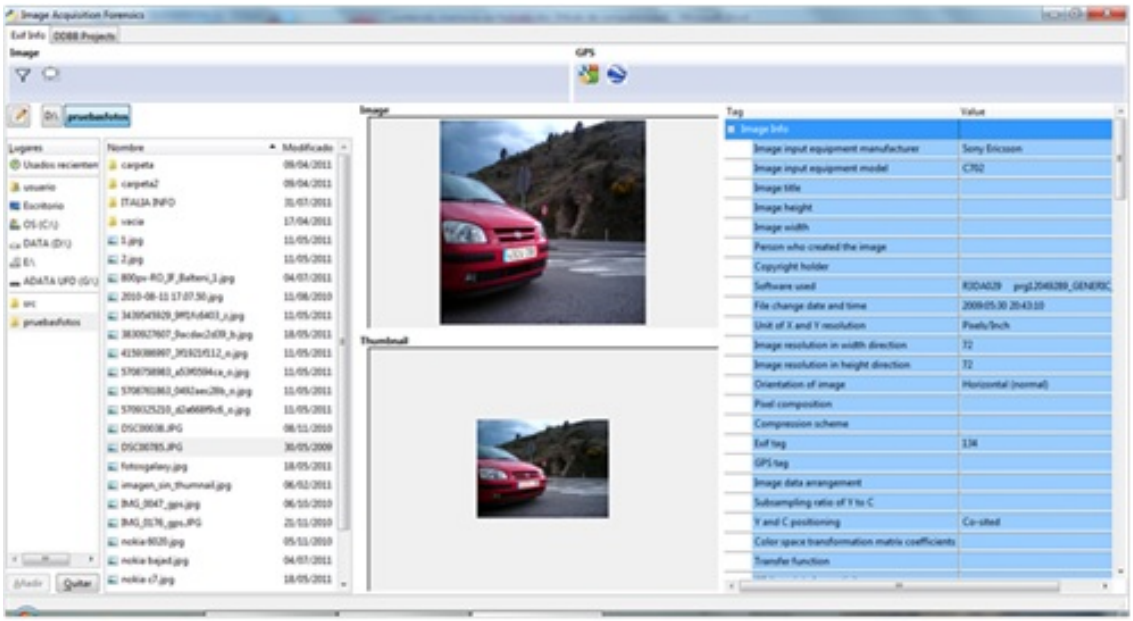


Figura A.1: Apariencia general de la pestaña Exif Info

Como estructura general se puede apreciar a la izquierda de la imagen un navegador de archivos, en el centro la imagen del archivo seleccionado y su correspondiente imagen en miniatura (es el incluido en el propio archivo de la imagen, no ninguna generación propia del programa) y a la derecha las etiquetas Exif con su correspondiente información. Cabe destacar que la interfaz gráfica es totalmente configurable a nivel de tamaños, es decir, todos los separadores entre las distintas zonas se pueden mover.

Una vez descrita la pantalla principal a grandes rasgos se van a presentar las opciones y funcionalidades.

- **Navegador de archivos**: Cuando un archivo de una imagen es seleccionado se muestra su imagen, su imagen en miniatura (si la posee) y toda la información Exif que ha podido ser extraída. Asimismo, si existe algún tipo de error en la apertura de la imagen, imagen en miniatura o el análisis de los datos Exif, los correspondientes apartados aparecen vacíos y se indica el error mediante el pertinente mensaje. Un ejemplo de apertura errónea de un archivo que no es una imagen se muestra en la Figura A.2.

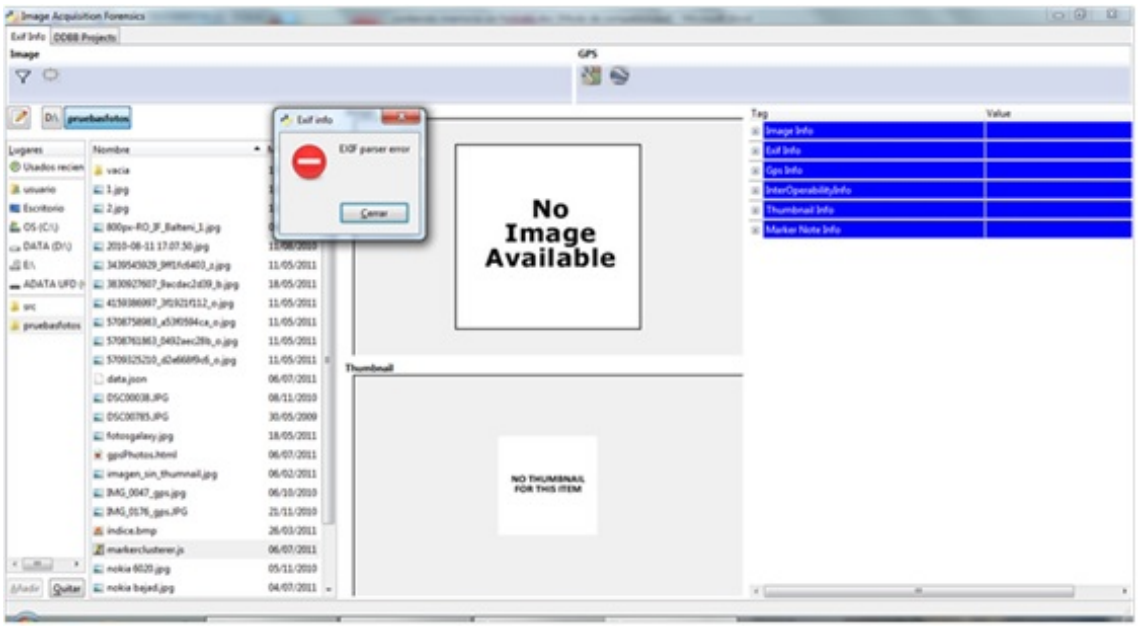


Figura A.2: Apertura errónea de un archivo

Además se permite almacenar y borrar los directorios de uso más común para facilitar el acceso a rutas. Para ello simplemente hay que seleccionar la ruta deseada y pulsar el botón "Añadir". Para eliminar la ruta almacenada sólo hay que seleccionarla y pulsar el botón "Quitar". Un ejemplo de esta funcionalidad se muestra en la Figura A.3.

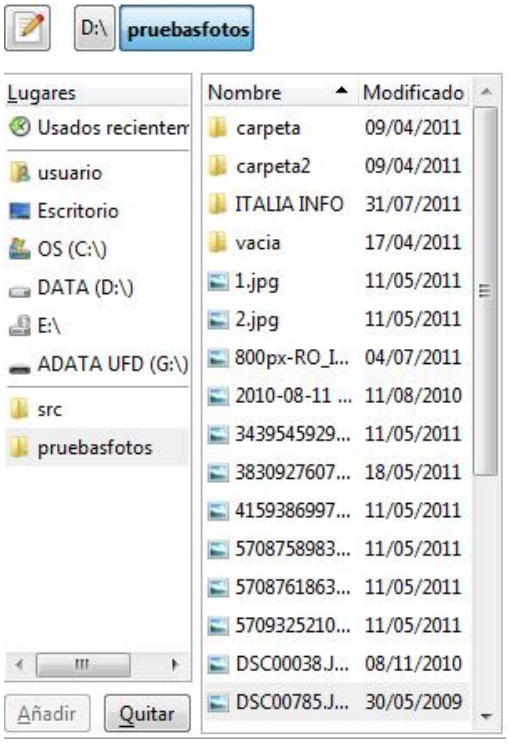


Figura A.3: Tratamiento de rutas

- **Menú Image**: Posee dos opciones, una para filtrar el tipo de archivos que se muestran en el navegador de imágenes y otra para cambiar el tamaño de la imagen y la imagen en miniatura a mostrar.

- **Menú GPS**: Se utiliza para sacar partido a los datos de geoposicionamiento que pueden ser incluidos en las imágenes. Si la imagen no tiene la suficiente información para poder ser mostrada en alguna de las opciones, al pulsarla se muestra un mensaje indicándolo.

  Dentro de este menú existen dos opciones: posicionamiento en *Google Maps* y en *Google Earth*. En la primera se abre el navegador web por defecto del sistema operativo y se muestra la ubicación inserta en los metadatos de la imagen en un mapa de *Google Maps* (es necesario conexión a Internet). La Figura A.4 muestra un ejemplo de geoposicionamiento en una fotografía en *Google Maps*.

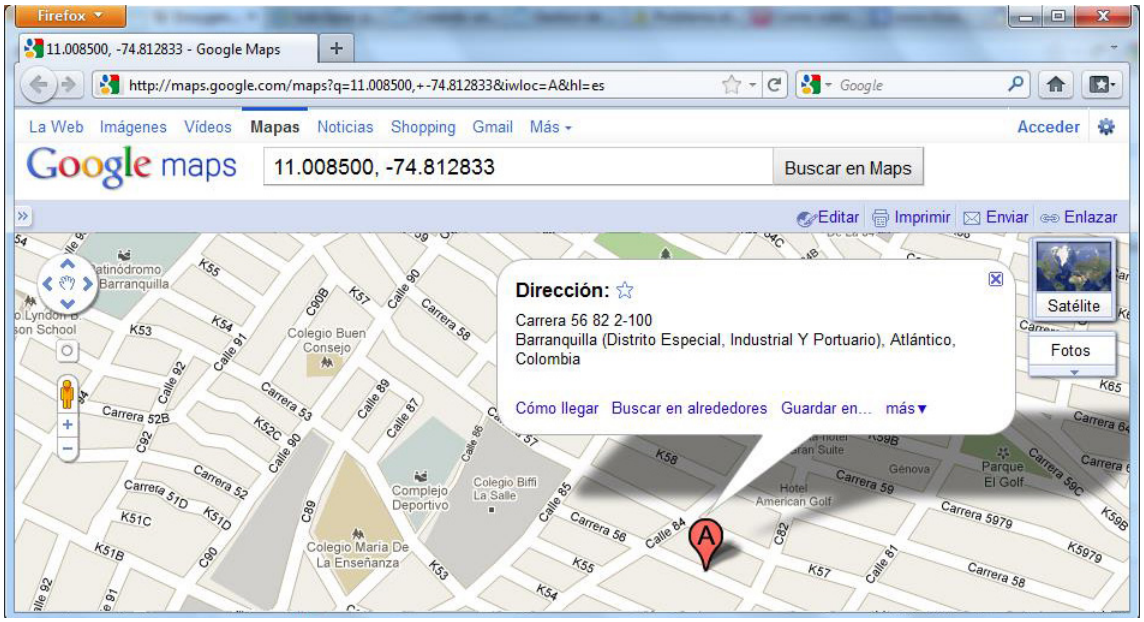


Figura A.4: Geoposicionamiento en Google Maps

En la segunda opción se abre un menú (Figura A.5) para poder almacenar un archivo de extensión "kml". Este archivo puede ser posteriormente abierto si está instalada la aplicación *Google Earth*, en la cual se muestra igualmente la posición geográfica almacenada en los metadatos de la imagen (es necesario conexión a Internet).

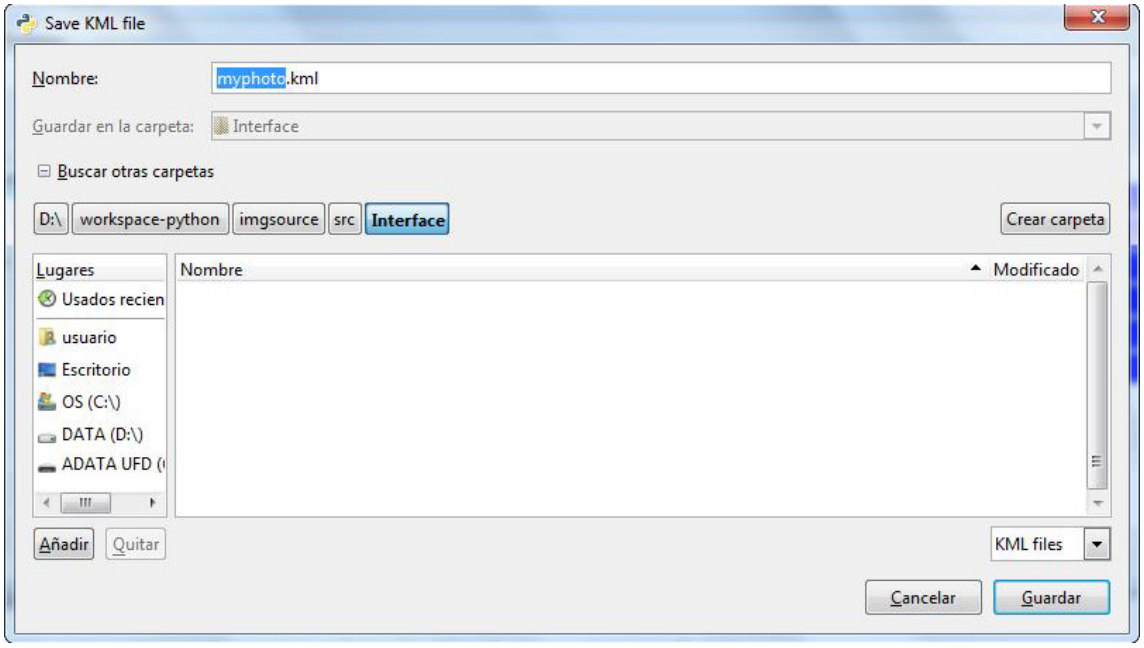


Figura A.5: Almacenamiento de archivos KML

La Figura A.6 muestra un ejemplo de geoposicionamiento de una fotografía en *Google Earth*.

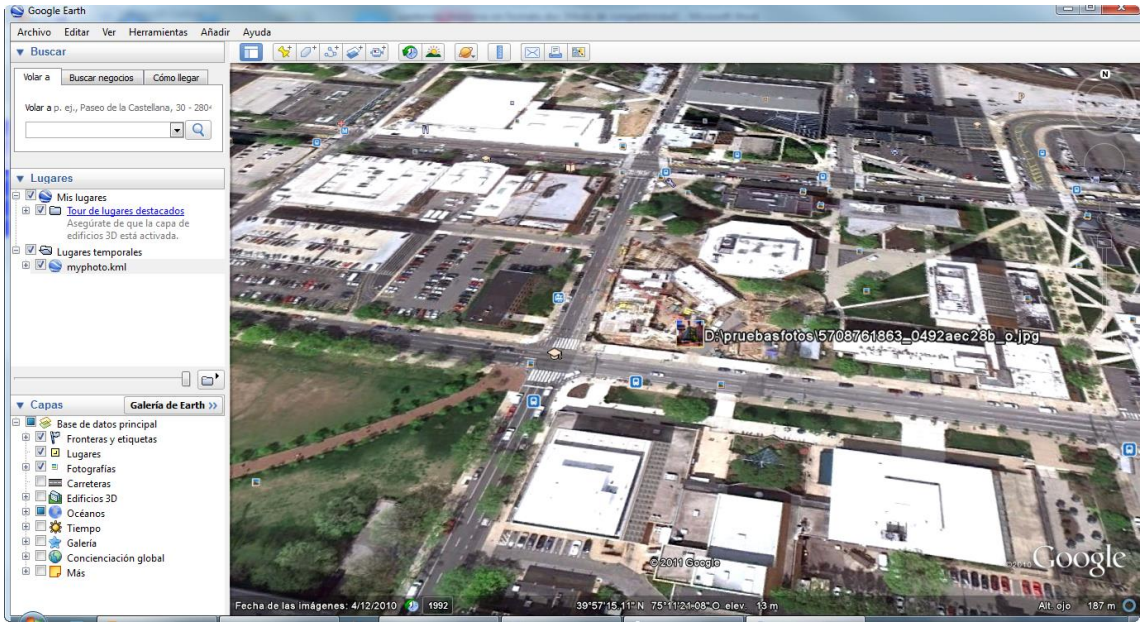


Figura A.6: Geoposicionamiento en Google Earth

- **Vista de imagen e imagen en miniatura**: Muestra la imagen y la imagen en miniatura del archivo seleccionado en el navegador de archivos. Este bloque posee barras de desplazamiento horizontal y vertical por si alguna de las imágenes es mayor que el tamaño que el usuario ha seleccionado para este espacio.

- **Etiquetas Exif**: En este bloque se muestran todos los metadatos Exif obtenidos del archivo seleccionado en el navegador de archivos. Se muestra la etiqueta y su correspondiente valor. Siempre se muestran todas las etiquetas que la aplicación captura; si una etiqueta no tiene valor para una imagen se muestra con valor vacío. Al pasar el ratón sobre el valor de una etiqueta aparece un menú contextual con una descripción orientativa basada en la propia especificación Exif 2.3. Como puede apreciarse en la Figura A.7, para mostrar la información se han creado 6 grupos: Image, Exif, GPS, Interoperability, Thumbnail y Maker Note.

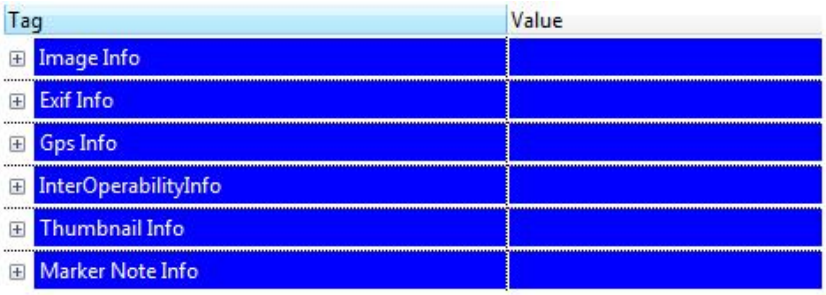


Figura A.7: Grupos de etiquetas Exif

A continuación se describe la información que aporta cada uno de los grupos:

- **Image Info**: En este bloque se almacenan las etiquetas con información relativa a la propia imagen y que no tienen relación directa con el entorno y el momento de la captura. Por ejemplo, la marca y el modelo de la cámara, el tamaño de la imagen, la unidad utilizada en la resolución X e Y, etc.

- **Exif Info**: En este bloque se guardan las etiquetas con información relativa al momento o al entorno de la toma de la imagen. Dentro de este bloque se encuentra por ejemplo, la información referente al flash, la hora de toma y generación de la imagen, la configuración de la lente, etc.

- **GPS Info**: En este bloque está toda la información relativa al geoposicionamiento. Por ejemplo, información de latitud, longitud, altitud, el estado del receptor GPS, etc.

- **InterOperability Info**: En este bloque se incluyen las etiquetas relativas a la información de las reglas de interoperabilidad, como pueden ser Exif R98, DCF thumbnail file o DCF Option file.

- **Thumbnail Info**: En este bloque se encuentran todas las etiquetas relativas a la información de la imagen en miniatura. Por ejemplo, su tamaño en vertical y horizontal y el esquema de compresión utilizado.

- **Maker Note Info**: Es una etiqueta individual que almacena la información que cada fabricante puede insertar de forma opcional y que no ha sido recogida en ninguna etiqueta Exif. El formato de esta información es libre y no tiene una estructura prefijada. Cada fabricante utiliza la suya propia que incluso puede ser diferente para distintos modelos de la misma marca. Por tanto, se muestra como una secuencia de bytes (en hexadecimal). Si se conoce la estructura, estos bytes pueden ser decodificados de forma manual.

A.2 Tratamiento de Imágenes a Nivel de Grupo

Esta funcionalidad está asociada a la pestaña DDBB Projects y su apariencia gráfica general puede verse en la Figura A.8.

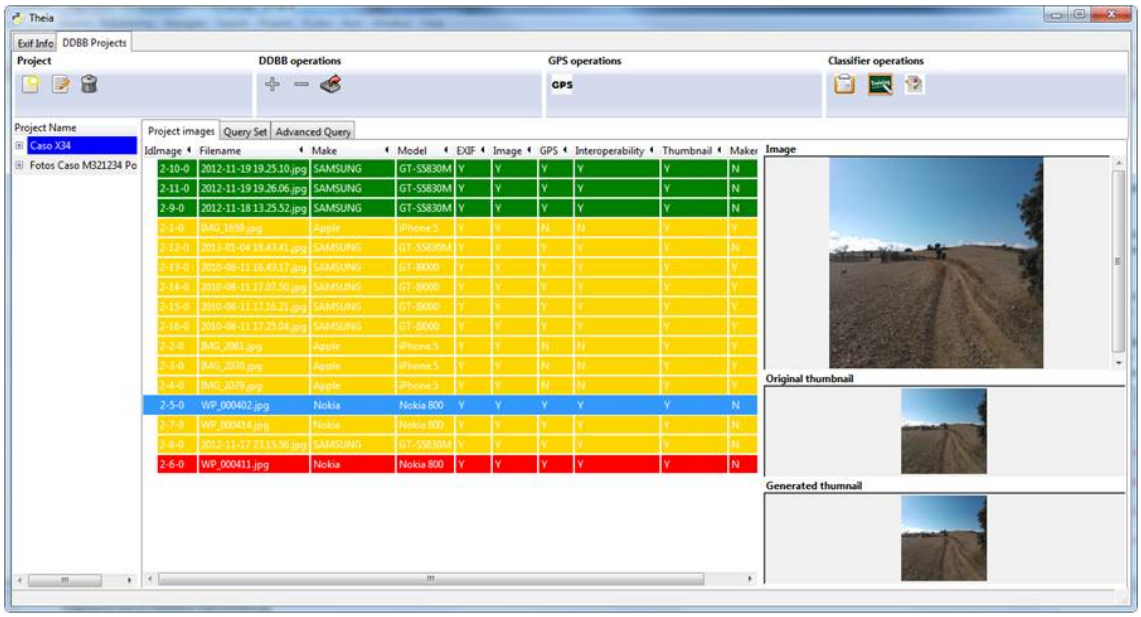


Figura A.8: Apariencia general de la pestaña DDBB Projects

La estructura de esta funcionalidad es mucho más compleja que la de la pestaña Exif Info. Asimismo, ofrece gran diversidad de opciones al analista forense.

Lo primero a destacar de esta funcionalidad es que las imágenes se tratan en grupos llamados proyectos. Estos grupos pueden ser de una o más imágenes. Cada proyecto es totalmente independiente entre sí. Se busca acercar la realidad del día a día del analista forense a la herramienta, es decir, el analista tendrá diversos casos de análisis disjuntos, los cuales podrá tratar en proyectos distintos.

Asimismo, en la parte central de la pantalla se muestran las imágenes que pertenecen al proyecto seleccionado. Para cada imagen se muestra información básica como el identificador de la imagen en la base de datos (para permitir el caso de archivos con el mismo nombre), el nombre del archivo, la marca y el modelo (obtenidos de los metadatos Exif). Además se presenta la información de si posee metadatos en los distintos grupos Exif que analiza la herramienta.

Cada fila de una imagen posee un color entre tres posibles: verde, amarillo o rojo. Estos colores indican el resultado del análisis de identificación de modificaciones en una imagen, comparando su imagen en miniatura con la generada por *Theia*, basándose en el

cálculo del RMS. El verde se corresponde con las imágenes que *Theia* clasifica como "no modificadas", el amarillo con las imágenes que clasifica como "posiblemente modificadas" y el rojo con las imágenes que clasifica como "modificadas".

Asimismo, a la izquierda de la pantalla se muestra la imagen seleccionada, la imagen en miniatura original almacenada en el archivo de la imagen (si la posee) y la imagen en miniatura generada por *Theia* con las librerías del lenguaje Python.

Dentro de esta pestaña se van a detallar las siguientes funcionalidades: gestión de proyectos, administración de imágenes de los proyectos, consultas sobre proyectos (*query set*), consultas avanzadas (*advanced query*), geoposicionamiento de las imágenes y técnicas de identificación de la fuente de adquisición.

A.2.1 Gestión de Proyectos

La gestión de proyectos se corresponde con el menú Project. Éste tiene las opciones de creación, edición y borrado de proyectos.

- **Creación de Proyectos**: Al pulsar sobre la creación de proyectos se abre una ventana como la de la Figura A.9. En ésta hay que introducir el nombre del proyecto, los formatos de archivos que se quieren incluir (todos los archivos, JPEG o *Bitmap* (BMP)) y la ruta de donde tomar los archivos. Se crea por tanto, un proyecto con el nombre indicado, que incluye todos los archivos filtrados según la opción de formato elegida que se encuentren en la ruta seleccionada y en todos sus subdirectorios (de cualquier nivel).

  Una vez introducido todos los parámetros de forma correcta se crea el nuevo proyecto mostrándose en el bloque de la izquierda. Además del nombre del proyecto, se muestra su fecha de creación y la carpeta base de donde se tomaron los archivos. Es muy importante destacar que en el momento de la creación del proyecto las imágenes se almacenan en una base de datos interna de la aplicación. Esta base de datos es totalmente independiente del directorio y no existe sincronización alguna con él; sólo se muestra la ruta de donde se cargaron los datos como información que puede ser de ayuda. Es decir, y haciendo hincapié, una vez creado el proyecto no hay relación alguna entre las imágenes de la base de datos de la aplicación y las imágenes del soporte físico de donde han sido extraídas.

  Cuando se pulsa el botón "aceptar" la aplicación va tomando uno a uno los archivos seleccionados y obteniendo sus metadatos para cargarlos en la base de datos interna. Asimismo, existen dos nuevas opciones relacionadas con las técnicas de identificación de la fuente de adquisición de las imágenes. Si está seleccionada la opción "*All*

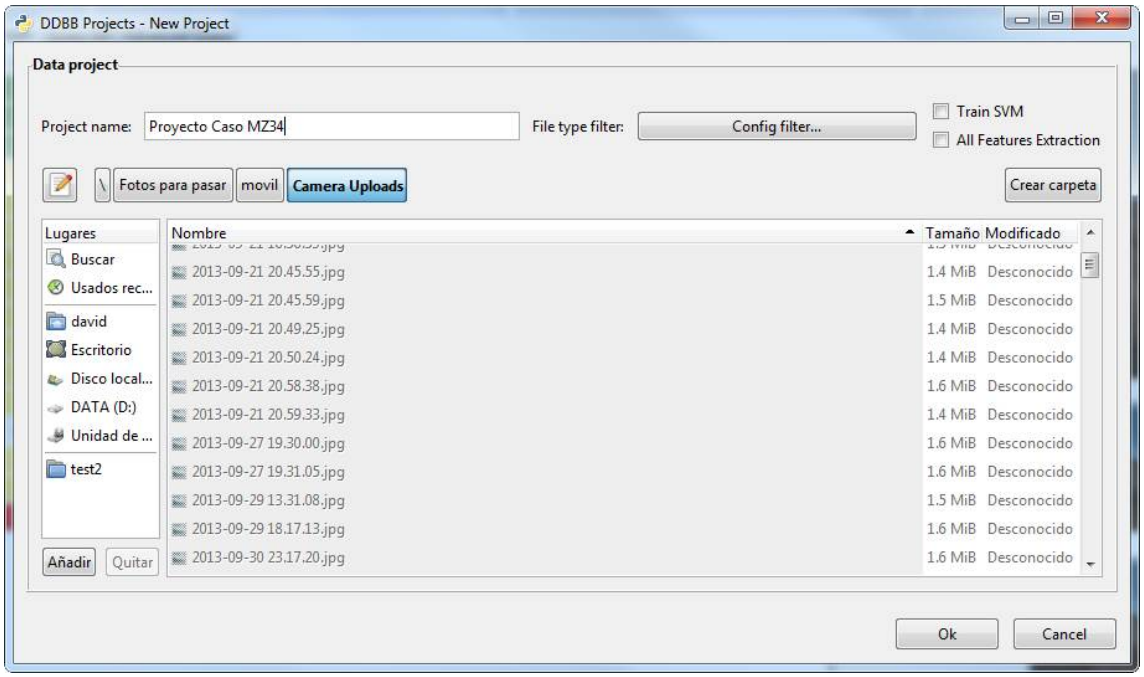


Figura A.9: Creación de proyectos

Features Extration", *Theia* extrae todas las características basadas en el contenido de la imagen definidas en la propia aplicación y las almacena en la base de datos. Si está seleccionada la opción "*Train SVM*" se realiza la fase de entrenamiento del clasificador SVM con el proyecto que se está creando. Los detalles de estas últimas dos opciones serán descritos en la Sección A.2.5

Si hay algún tipo de problema en el tratamiento de los archivos (archivos que no son imágenes, problemas con los permisos del sistema operativo, errores en el análisis sintáctico, etc.), la aplicación genera una lista con los archivos que no han podido ser incluidos en la base de datos y la razón de su no inclusión. Si todos los archivos se han incluido en la base de datos, no se muestra lista alguna. En la Figura A.10 se muestra un ejemplo de la forma de presentar la información de un proyecto.

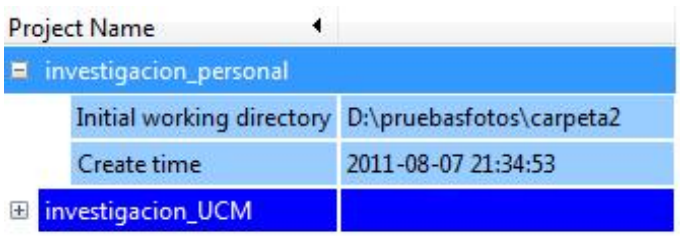


Figura A.10: Información de proyectos

- **Edición de Proyectos**: Permite editar el nombre del proyecto. La Figura A.11 muestra la pantalla de edición de proyectos.

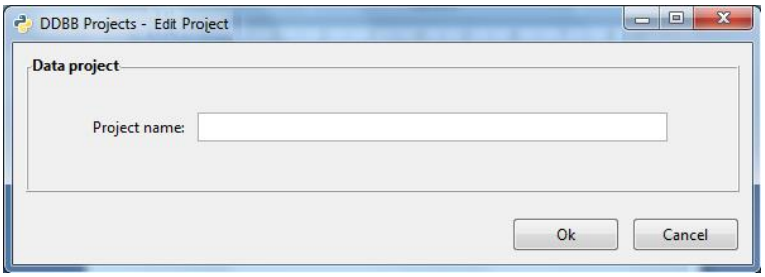


Figura A.11: Edición de proyectos

- **Borrado de Proyectos**: Permite eliminar el proyecto seleccionado. Se muestra un mensaje de confirmación antes de realizarse el borrado final. Una vez borrado el proyecto es imposible recuperarlo.

A.2.2 Administración de Imágenes de los Proyectos

La administración de imágenes de cada uno de los proyectos se realiza con el menú DDBB Operations. Dentro de este menú se encuentran las opciones de añadir y eliminar imágenes de un proyecto, visualización de las imágenes de un proyecto y exportarlas a un directorio.

- **Añadir Imágenes a un Proyecto**: El añadir imágenes a un proyecto es análogo a la creación de un nuevo proyecto, salvo que (como es evidente) el nombre del proyecto no se puede editar. Cabe destacar que en un mismo proyecto pueden haber dos imágenes con el mismo nombre y contenido. Es decir, en un proyecto puede estar el mismo archivo incluido varias veces. Este caso se permite, ya que un analista forense puede querer examinar un dispositivo y en éste puede estar el mismo archivo repetido varias veces en distintas ubicaciones. La Figura A.12 muestra la pantalla para añadir imágenes a un proyecto.

  Si durante el proceso se presentan errores, se insertan en el proyecto creado las fotografías que están correctas y se muestra un informe de los ficheros que no se pueden agregar al proyecto y la causa para cada uno de ellos.

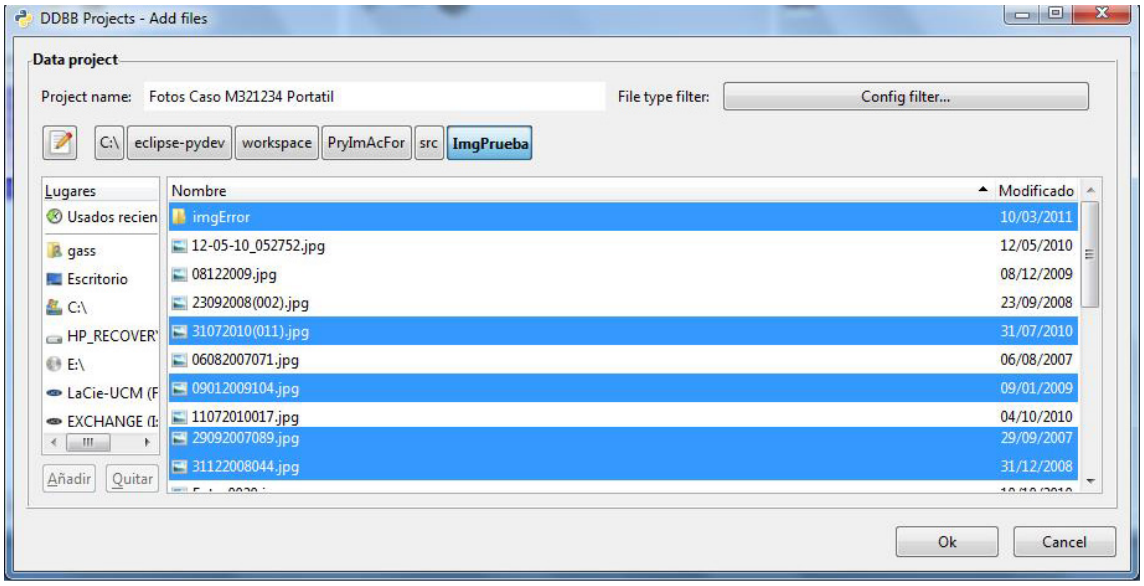


Figura A.12: Añadir imágenes a proyectos

- **Eliminar Imágenes de un Proyecto**: Al pulsar este botón se abre una ventana con una lista de todas las imágenes del proyecto para poder seleccionar una o varias imágenes y proceder a su eliminación. Si se seleccionan las imágenes una a una, además se muestra el contenido de la misma. La Figura A.13 muestra la pantalla de eliminación de imágenes de un proyecto.

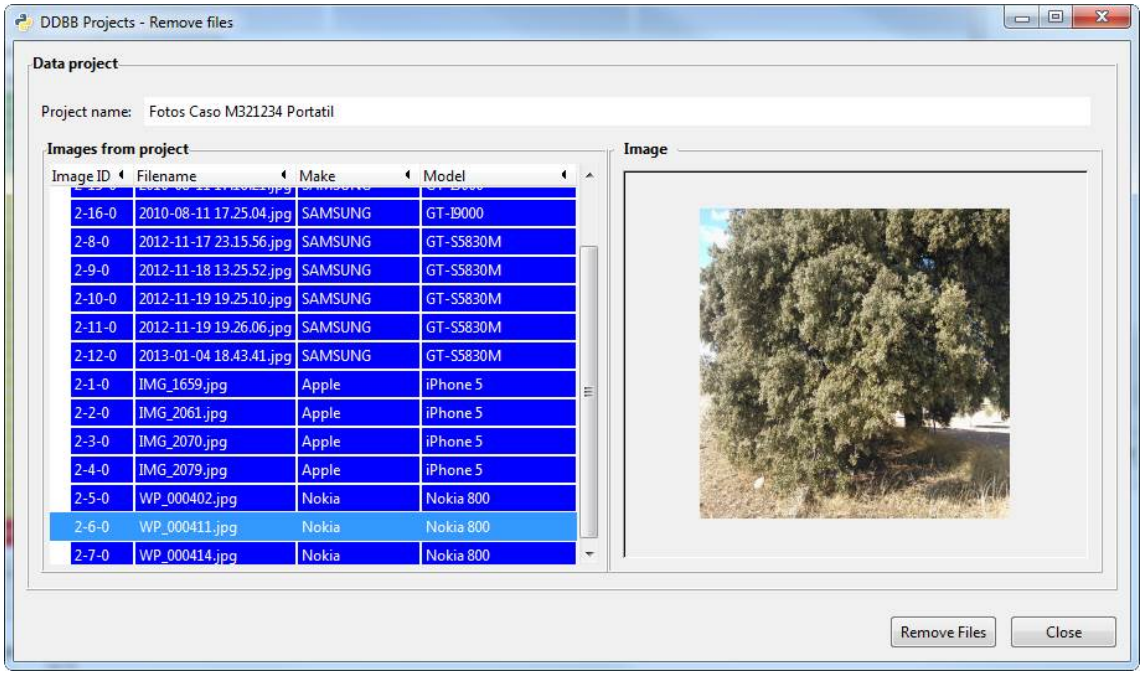


Figura A.13: Eliminar imágenes de proyectos

- **Visualización de las imágenes de un proyecto**: En la parte central de la pestaña DDBB Projects y, dentro de ésta, en la pestaña Project Images se muestra una lista de las imágenes del proyecto seleccionado en la lista de proyectos.

  Para cada imagen se muestra su identificador interno de la base de datos (para permitir el caso de archivos con el mismo nombre), el nombre del archivo y la marca y el modelo de dispositivo que la generó (obtenidos de los metadatos Exif). Además, se presenta la información de si posee metadatos en los distintos grupos Exif que analiza la herramienta. Asimismo, se visualiza a la izquierda según se van seleccionando el contenido de cada una de las imágenes, la imagen en miniatura original almacenada en el archivo de la imagen (si lo posee) y la imagen en miniatura generada por *Theia* con las librerías del lenguaje Python.

  Como se comentó anteriormente, cada fila de una imagen posee un color entre tres posibles: verde, amarillo o rojo. Estos colores indican el resultado del análisis de identificación de modificaciones en una imagen, comparando su imagen en miniatura con la generada por *Theia* basándose en el cálculo del RMS. El verde se corresponde con las imágenes que *Theia* clasifica como "no modificadas", el amarillo con las imágenes que clasifica como "posiblemente modificadas" y el rojo con las imágenes que clasifica como "modificadas". Un ejemplo de captura de esta funcionalidad se muestra en la Figura A.14.

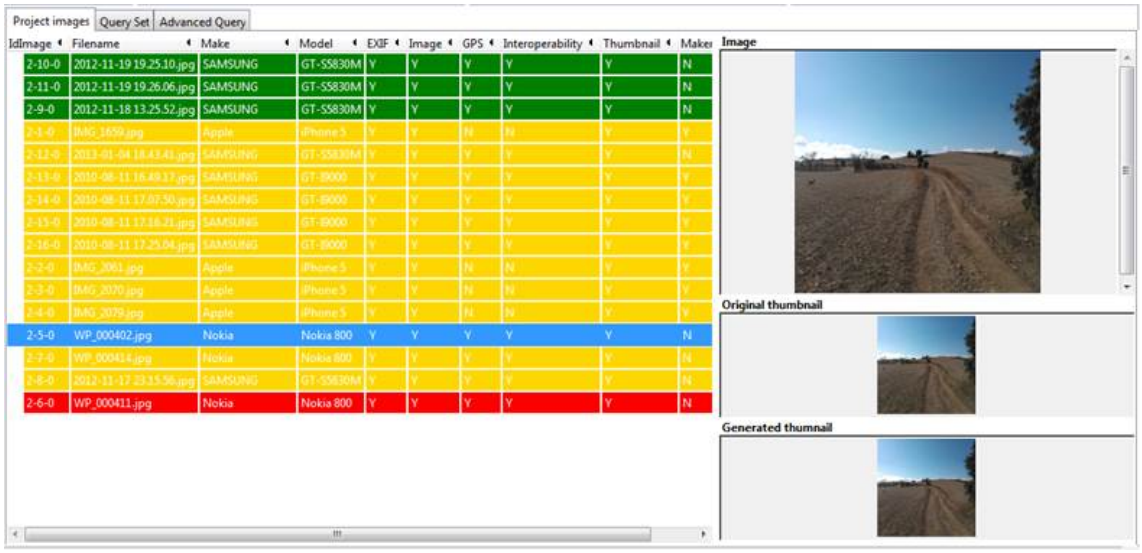


Figura A.14: Visualización de las imágenes de un proyecto

- **Exportar las imágenes de un proyecto**: Esta opción permite exportar un grupo de archivos de un proyecto a una ruta seleccionada por el usuario.

A.2.3 Consultas en Conjunto

La funcionalidad de consultas en conjunto se encuentra en la pestaña Query Set de la pestaña principal DDBB Projects. En esta opción se permite crear consultas agregando etiquetas Exif (y otras adicionales que añade la aplicación) sobre las imágenes del proyecto seleccionado. Un ejemplo de apariencia general puede verse en la Figura A.15.

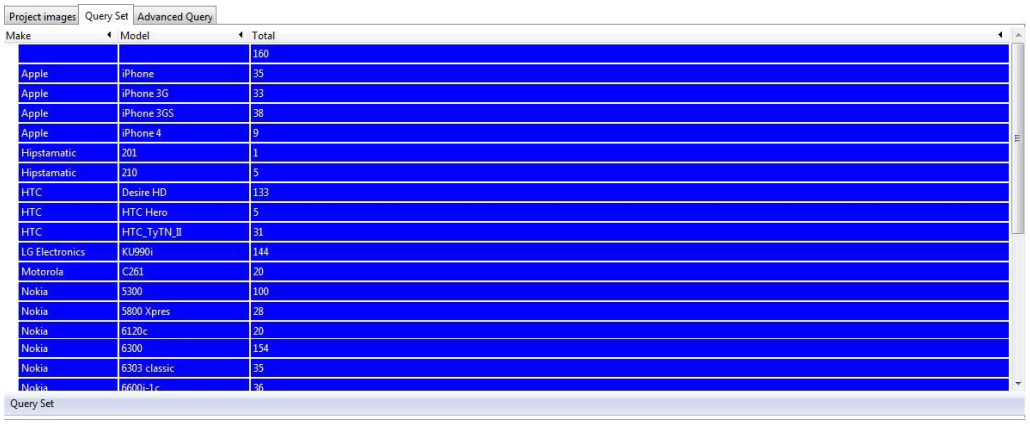


Figura A.15: Consultas en conjunto

En las consultas permiten elegir 5 campos de agregación como máximo (por defecto se realiza sobre Make y Model, aunque puede ser cualquiera de la lista). Para elegir los distintos campos hay que pulsar sobre el botón "Query Set" y aparece la ventana de la Figura A.16.

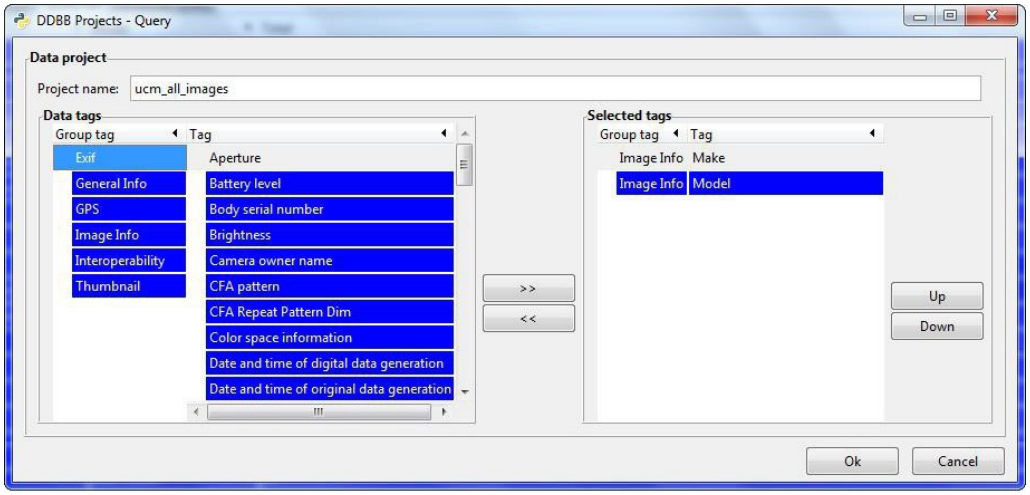


Figura A.16: Selección de campos de agregación

En la Figura A.16 se puede elegir cualquiera de las etiquetas Exif de los distintos

grupos que trata la aplicación (Exif Info, General Info, GPS, Image Info, Interoperability y Thumbnail). Además se puede elegir un grupo adicional de información general, en el cual se han incluido nuevos campos que se consideran interesantes para este tipo de consultas.

Los campos incluidos en el grupo General Info son: fecha de creación de la imagen, ruta de origen de la carga de la imagen, identificador interno de la base de datos, nombre de archivo, proyecto al que pertenece, formato del archivo y si posee información para cada uno de los grupos Exif Info, GPS Info, Image Info, Interoperability Info, Maker Note y Thumbnail. Una vez elegidos los campos, se puede modificar el orden en el que quieren ser mostrados en el resultado. Para ello se utilizan los botones "Up" y "Down". Finalmente, para ejecutar la consulta pulsar "Ok" y aparece el resultado en la ventana inicial de Query Set.

La consulta agrupa las imágenes por los criterios seleccionados y muestra el número de imágenes que hay en cada uno de los grupos formados. Por ejemplo, si se quiere ver cuántas imágenes hay de cada marca y modelo en un proyecto se deben seleccionar los campos "Image input equipment manufacturer" e "Image input equipment model" del grupo "Image Info" y se pulsa "Ok". El resultado muestra todas las marcas y modelos de dispositivos móviles que hay en ese proyecto y el número de imágenes de cada uno (ver Figura A.15).

En la funcionalidad de consultas avanzadas (Advanced Query) hay que distinguir dos grandes bloques: la configuración de la consulta y su almacenamiento. Con respecto a la configuración de la consulta avanzada hay que tener en cuenta la configuración de las columnas de los resultados y la configuración de los filtros. En esta consulta se muestran los valores de los campos seleccionados por la configuración de las columnas de los resultados que cumplen las restricciones indicadas en la configuración de los filtros.

- **Configuración de las columnas de los resultados**: Esta opción se realiza con el botón "Config Query Columns" del menú "Config". Muestra una ventana para la selección de los campos que se quieren mostrar como columnas en el resultado de la consulta. Esta ventana tiene el mismo modo de funcionamiento que la utilizada en Query Set salvo con la excepción de no tener límite para el número de campos que se pueden elegir en las columnas. Al menos una columna debe ser elegida antes de ejecutar la consulta. En caso contrario, la aplicación lo indica con un mensaje de error. Una vez seleccionados estos campos se muestran como columnas en la parte superior como se indica en la Figura A.17.

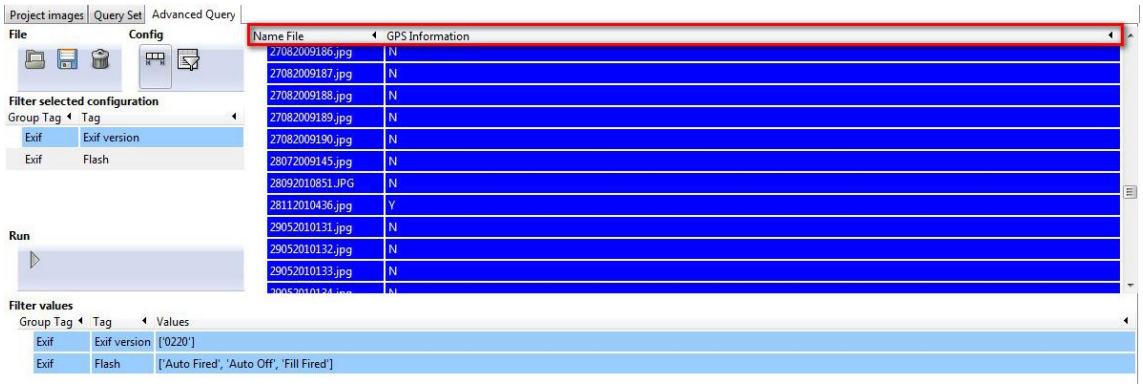


Figura A.17: Configuración de las columnas de resultado

- **Configuración de los filtros**: Para configurar esta opción hay que realizar varios pasos:

    1. Seleccionar los campos que se utilizarán como filtros: La ventana de selección de estos campos es análoga a la pantalla de selección de campos para la configuración de las columnas. Una vez elegidos los filtros se incluyen en el bloque " Filter selected configuration" como se aprecia en la Figura A.18.

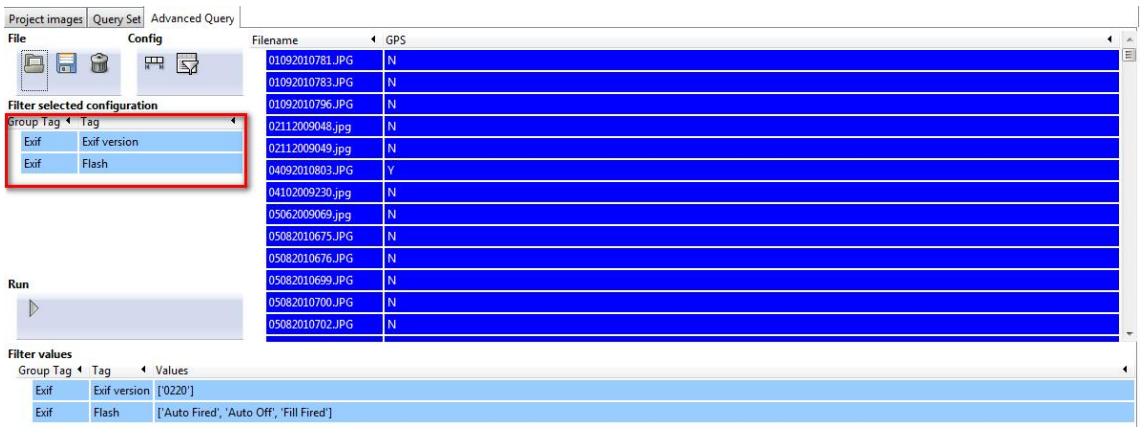


Figura A.18: Configuración de filtros

    2. Configurar los valores de cada uno de los filtros para ejecutar finalmente la consulta: Si existe algún filtro sin configurar a la hora de ejecutar la consulta se muestra un mensaje de error indicándolo. Para configurar cada uno de los filtros hay que hacer doble clic en cada uno de ellos en el bloque "Filter selected configuration". Tras el doble clic aparece una ventana con los valores posibles de esa etiqueta para las imágenes del proyecto seleccionado. El analista debe seleccionar los valores por los que se desea realizar el filtrado. Si seleccionan varios valores para un mismo filtro, la imagen debe poseer uno de los valores

pero no todos, ya que en Exif cada campo sólo puede tener un valor y no un conjunto de valores. Por ejemplo, si se elige para filtrar el campo "Exif Flash" y se configuran los valores "Auto Fired" y "Auto Off", al ejecutar la consulta, se muestran los valores de la columnas configuradas de las imágenes que posean en la etiqueta flash el valor "Auto Fired" o "Auto Off". Por tanto, en la configuración de los valores de cada uno de los filtros, la aplicación al realizar la consulta hace una O lógica (or) con respecto a los valores seleccionados. En cambio, entre los distintos filtros seleccionados la aplicación al realizar la consulta hace una Y lógica (and) entre los distintos campos a filtrar.

Una vez configurada totalmente la consulta para ejecutarla hay que pulsar el botón "Run query to current project" y seguidamente se muestran los resultados.

Por ejemplo, si se quiere obtener el nombre de la imagen y el valor de la latitud de las imágenes que han sido realizadas con flash "Auto Fired" o "Fired" y que además tengan información en algunos de sus campos GPS hay que realizar los siguientes pasos:

1. Configurar las columnas de resultados eligiendo los campos "Name File" y "Latitude".
2. Seleccionar los filtros Exif Flash y General Info GPS Information.
3. Configurar los valores de los filtros seleccionados. Para "Exif Flash" tomar los valores "Auto Fired" y "Fired" y para "General Info GPS Information" el valor "Y".
4. Pulsar el botón "Run query to current project".

El resultado obtenido para la anterior consulta y un proyecto de prueba se muestra en la Figura A.19.

Se puede observar que existen 4 imágenes que cumplen los criterios y se muestran sus latitudes ([grados, minutos, segundos]). En la segunda imagen de la lista se puede apreciar que no hay información de latitud. Inicialmente esto puede chocar un poco, ya que uno de los filtros indicaba que tenía que tener información GPS. El resultado es bueno ya que esa imagen posee información GPS en otras etiquetas pero posee la etiqueta de la latitud vacía.

La aplicación también permite el almacenamiento de las consultas. El fin de esta funcionalidad es la de poder almacenar consultas en las que se invierte una cantidad considerable de tiempo para configurarlas y posteriormente poder utilizarlas en distintas ejecuciones de la herramienta. Para ello se utilizan los botones del menú "File" el cual permite abrir, guardar y borrar una consulta avanzada.

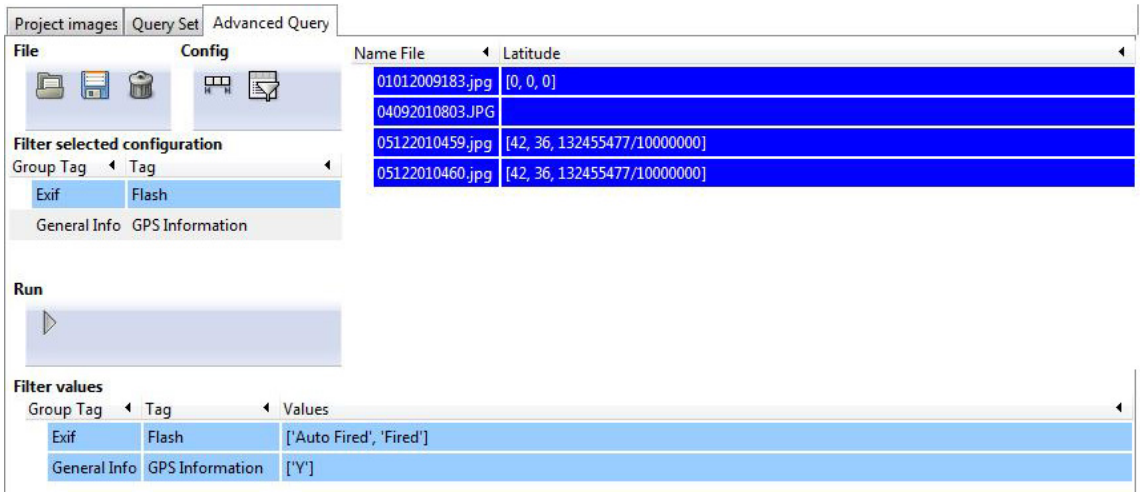


Figura A.19: Ejemplo de resultados de consulta con Advanced Query

- **Guardar una consulta avanzada**: Pulsando el botón "Save Advanced Query" se permite el almacenamiento permanente de una consulta. Aparece una ventana de diálogo donde se introduce el nombre que se desea poner a la consulta a almacenar (debe ser único). Si se utiliza un nombre de una consulta existente la herramienta avisa al usuario de esa situación y pide confirmación para sobrescribir la existente. Cabe destacar que para almacenar una consulta avanzada no tiene por qué estar totalmente configurada. La consulta se almacena en la base de datos para el proyecto que esté seleccionado.
- **Apertura de una consulta avanzada almacenada**: Para abrir una consulta guardada hay que pulsar el botón "Open Advanced Query", que hace que se muestre una ventana con una lista de las consultas almacenadas para el proyecto seleccionado.
- **Borrar una consulta almacenada**: Al pulsar sobre el botón "Delete Advanced Query" se abre una ventana con la lista de las consultas almacenadas para el proyecto seleccionado. Antes de su borrado final de la base de datos de la aplicación se pide confirmación al usuario. Una vez borrada una consulta es irrecuperable.

A.2.4 Geoposicionamiento

Al igual que con el tratamiento de imágenes individual existe una funcionalidad que permite el tratamiento de la información de geoposicionamiento para las imágenes de un proyecto. Ésta se encuentra en el botón "GPS position of project files" del menú "GPS operations". Una vez pulsado este botón se abre una ventana como la de la Figura A.20 con las siguientes partes:

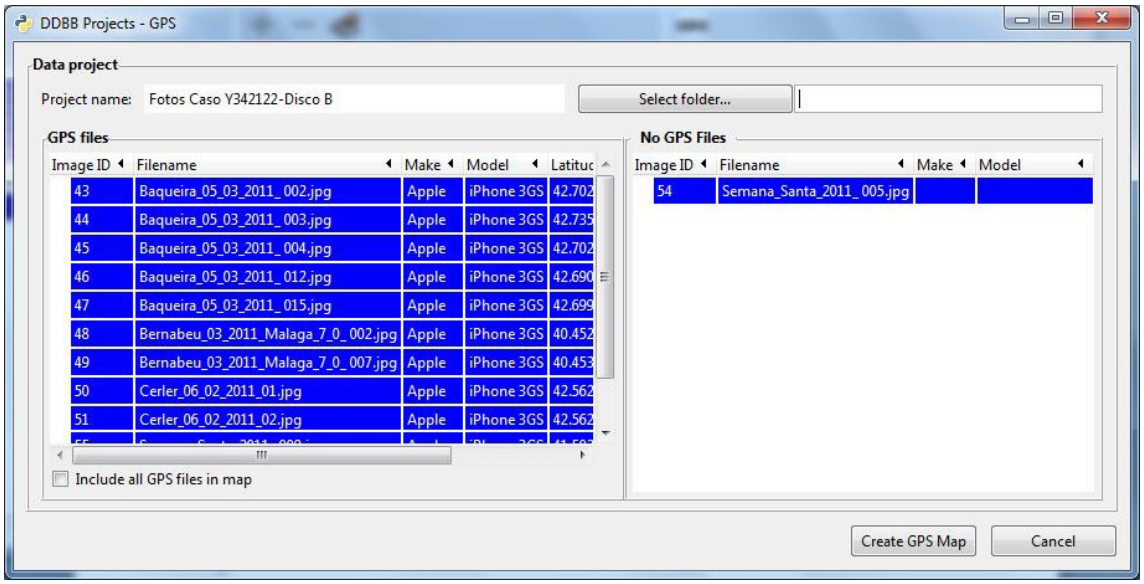
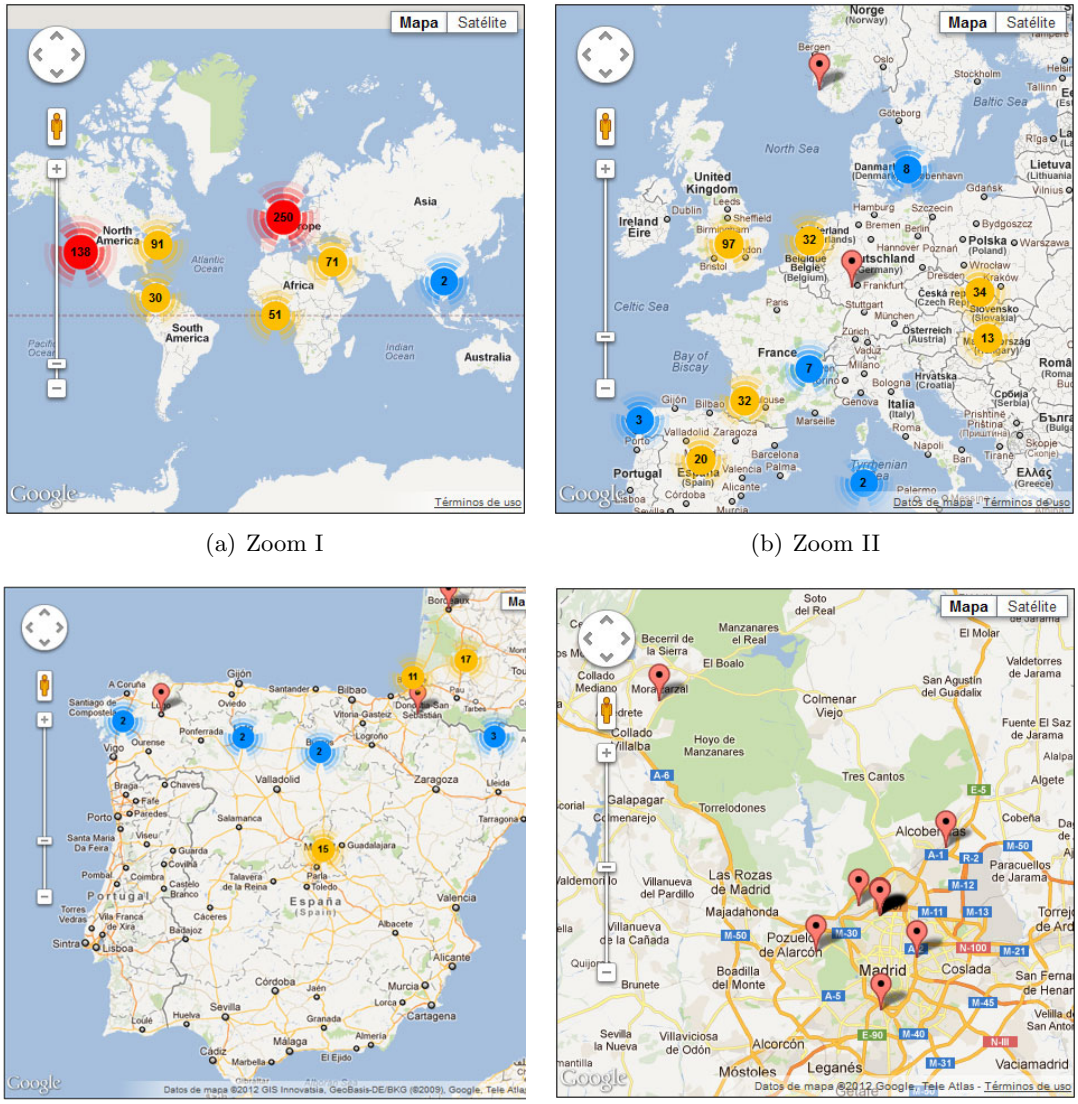


Figura A.20: Geoposicionamiento

- **Nombre del proyecto actual**: Campo no editable.
- **Lista de imágenes con información GPS de latitud y longitud**: De esta lista el usuario puede seleccionar las imágenes que se quieren que sean ubicadas en el mapa. Si se quiere que sean todas las del proyecto seleccionar la opción "Include all GPS files in map".
- **Lista de imágenes sin información GPS de latitud y longitud**: Sólo se muestran a nivel informativo para que el usuario sea consciente y pueda ver las imágenes que no pueden ser ubicadas en el mapa por no tener suficiente información de geoposicionamiento.
- **Selección de ruta donde se almacenarán los archivos de los mapas**: Es obligatorio que se seleccione una ruta donde almacenar físicamente los archivos de los mapas a generar. Esto permite que se puedan crear varios mapas con distintas imágenes de un mismo proyecto. Además permite portar los archivos y poder ser visualizados sin la herramienta. En la carpeta seleccionada se guardan dos archivos auxiliares (data.json y markerclusterer.js) y un archivo HTML (gpsPhotos.html) el cual es el que se debe abrir con un navegador web para mostrar el mapa. Para la visualización del mapa es necesario tener conexión a Internet.
- Botón **"Create GPS Map"**: Al pulsar este botón y estar configurados correctamente todos los parámetros anteriormente citados se crean los archivos del mapa en la ruta especificada y se lanza la visualización del mapa en el navegador por defecto. En el mapa se agrupan las fotos por zona, y a medida que se aumenta el zoom se van detallando las coordenadas de los grupos de imágenes

o imágenes individuales. La Figura A.21 muestra un ejemplo del mapa generado y el proceso de aumento del zoom en una zona concreta (desde la Figura A.21(a) hasta la Figura A.21(d)).



(a) Zoom I

(b) Zoom II

(c) Zoom III

(d) Zoom IV

Figura A.21: Geoposicionamiento de un grupo de imágenes en Google Maps

A.2.5 Identificación de la Fuente de Adquisición de Imágenes

Theia permite la identificación de la fuente de adquisición de un conjunto de imágenes de un proyecto. Las técnicas utilizadas se basan en las características del contenido de la imagen extraídas con los algoritmos propuestos en esta tesis. Asimismo la configuración de los distintos parámetros de la máquina SVM son los descritos en esta Tesis.

Las funciones de *Theia* con respecto a la identificación de la fuente de adquisición se controlan con los botones que se muestran en la Figura A.22, los cuales pertenecen a la pestaña DDBB Projects. De izquierda a derecha en la Figura A.22 se muestran los botones para la extracción de características, entrenamiento de la máquina SVM y clasificación con la máquina SVM, respectivamente.

Figura A.22: Botones para la identificación de la fuente de adquisición de imágenes

A continuación se detallan las funcionalidades de Theia con respecto a la identificación de la fuente de adquisición de imágenes.

- **Extracción de características**: Para cualquier operación relacionada con la identificación de la fuente de adquisición en *Theia*, es necesario que las características de las imágenes a utilizar hayan sido previamente extraídas. Existen dos formas de extracción de características de las imágenes de un proyecto: en el momento de su creación (descrito en la Sección A.2.1) o mediante la ventana de extracción de características (ver Figura A.23).

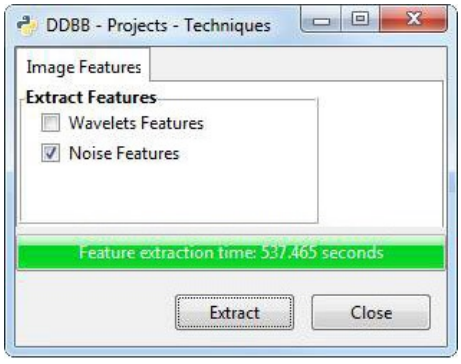


Figura A.23: Extracción de características

Como se observa en la Figura A.23 *Theia* permite seleccionar al usuario los conjuntos de características que se quieren extraer. Estos conjuntos de características se corresponden con las descritas en los algoritmos de extracción propuestos en esta Tesis. "Wavelets Features" se corresponden con las características basadas en el patrón del ruido del sensor (PRNU) descritas en el Capítulo 9 y "Noise Features" se

corresponden con las características del ruido del sensor descritas en el Capítulo 8.

Al pulsar el botón "*Extract*" se extraen las características seleccionadas y se almacenan en la base de datos. Asimismo, se muestra una barra con el progreso de la operación y cuando éste termina, se presenta el tiempo de ejecución utilizado para la extracción de características. Cuando de una imagen se extrae un conjunto de características y se almacena en la base de datos, no se vuelve a extraer en procesos futuros. *Theia* controla este hecho con el objetivo de optimizar el tiempo de extracción.

Cabe destacar que las características extraídas se almacenan en la base de datos de forma permanente, estando disponibles para el analista forense en distintas ejecuciones de *Theia*, ya que los procesos de extracción de características de un gran número de imágenes pueden ser costosos en tiempo.

- **Entrenamiento de la máquina SVM**: Tras la extracción de las características, para el proceso de identificación de la fuente de la adquisición de la imagen es necesaria la fase de entrenamiento del clasificador SVM. La ventana de la configuración de la fase de entrenamiento se muestra en la Figura A.24.

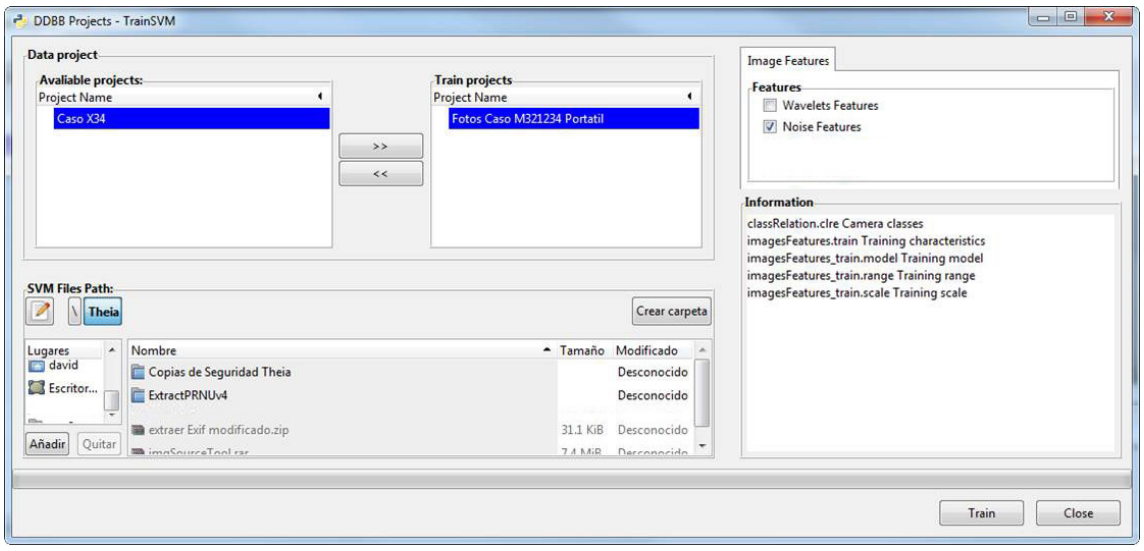


Figura A.24: Configuración de la fase de entrenamiento de la máquina SVM

Los puntos a tener en cuenta a la hora de la configuración de la fase de entrenamiento de la máquina SVM son los siguientes:

  - Selección de proyectos para el entrenamiento: *Theia* permite seleccionar tantos proyectos como se deseen para entrenar el clasificador SVM. Tiene que haber al menos uno seleccionado.

– Selección de proyectos para el entrenamiento: *Theia* permite seleccionar el conjunto de características a utilizar en la fase de entrenamiento del clasificador SVM. Es necesario que los conjuntos de características seleccionados hayan sido previamente extraídos en las imágenes de los proyectos. Si no se han extraído las características necesarias de los proyectos seleccionados para realizar la fase de entrenamiento, *Theia* avisa con un mensaje de advertencia indicando tal circunstancia (ver Figura A.25).

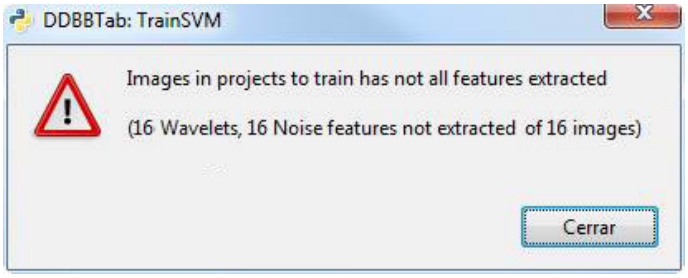


Figura A.25: Mensaje de error en la fase de entrenamiento

– Selección de proyectos para el entrenamiento: En la fase de entrenamiento el clasificador SVM genera un conjunto de archivos necesarios para la fase de clasificación. Estos archivos son almacenados en la ruta que indique el usuario. Asimismo, los archivos con los resultados de la fase de clasificación son almacenados en esta misma ruta.
– Información sobre la fase de entrenamiento: Se muestra información textual estática sobre los archivos generados en el proceso de entrenamiento. De los cinco archivos generados se destacan por su importancia dos: *classRelation.clre* e *imageFeatures.train*. El archivo *classRelation.clre* almacena la correspondencia entre los identificadores numéricos internos que trata el clasificador SVM y las cadenas de caracteres de la marca y el modelo de los dispositivos utilizados en la fase de entrenamiento. El archivo *imageFeatures.train* almacena para cada imagen utilizada en la fase de entrenamiento el identificador numérico interno de la máquina SVM y los valores de cada una de las características seleccionadas.

Una vez configurados todos los parámetros, el usuario debe pulsar el botón "Train" para comenzar con la fase de entrenamiento. Se muestra una barra con el progreso del proceso y a su finalización se muestra el tiempo invertido. Una vez finalizado el proceso esta ventana puede cerrarse utilizando el botón "Close".

- **Clasificación con la máquina SVM**: Una vez realizada la fase de entrenamiento se puede comenzar con la fase de clasificación, la cual es la que finalmente identifica la fuente de las imágenes utilizando las características seleccionadas por el usuario.

  Cabe destacar que para realizar la clasificación de las imágenes de un proyecto es necesario que estén extraídos los conjuntos de características con los que entrenó al clasificador SVM. De no estar extraídas al menos las características necesarias, *Theia* muestra un mensaje advirtiendo esta circunstancia para que el usuario extraiga las características necesarias de las imágenes del proyecto.

  En la fase de clasificación el usuario sólo puede utilizar las imágenes de un proyecto para que se identifique su fuente de adquisición. El proyecto seleccionado en la pestaña DDBB Projects, es el utilizado para la fase de clasificación. Una vez seleccionado el proyecto el usuario debe pulsar el botón de clasificación para iniciar el proceso de identificación de la fuente de adquisición de las imágenes del proyecto seleccionado. *Theia* muestra una ventana con el progreso del proceso y finalizado el mismo presenta una ventana indicando el tiempo de ejecución consumido (ver Figura A.26).

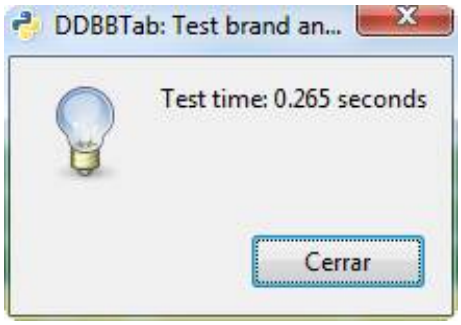


Figura A.26: Mensaje de finalización del proceso de clasificación

Una vez terminado el proceso de clasificación de la fuente por parte de la máquina SVM se generan dos archivos: *imagesFeatures.test* e *imagesFeatures_test.predict*. *ImagesFeatures.test* almacena para cada imagen utilizada en la fase de clasificación el identificador numérico interno de la máquina SVM y los valores de cada una de las características seleccionadas; *imagesFeatures_test.predict* es el que almacena los resultados de la identificación de la fuente de adquisición de las imágenes del proyecto. Para cada imagen del archivo *imageFeatures.train* en *imagesFeatures_test.predict* se muestra la predicción de la marca y el modelo realizada por *Theia*. Dicho de otro modo, *imagesFeatures_test.predict* almacena el resultado del todo el proceso de identificación de la fuente de adquisición del proyecto seleccionado.

Parte IV

Publicaciones

Apéndice B

Lista de Publicaciones

A continuación se muestran los trabajos que se derivan a la realización de la presente Tesis Doctoral:

- Ana Lucila Sandoval Orozco, David Manuel Arenas González, Luis Javier García Villalba, Julio César Hernández Castro: Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles. Actas de la XII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2012), Donostia-San Sebastián, España, Septiembre 4 – 7, 2012.
- Ana Lucila Sandoval Orozco, David Manuel Arenas González, Luis Javier García Villalba, Julio César Hernández Castro: Análisis Forense de Imágenes de Dispositivos Móviles Utilizando los Metadatos Exif. Actas del XXVII Simposium Nacional de la Unión Científica Internacional de Radio (URSI 2012), Elche, Alicante, España, Septiembre 12 – 14, 2012.
- Ana Lucila Sandoval Orozco, Jocelin Rosales Corripio, David Manuel Arenas González, Luis Javier García Villalba, Julio César Hernández Castro: Techniques for Source Camera Identification. Proceedings of the 6th International Conference on Information Technology (ICIT 2013), Amman, Jordan, May 8 – 10, 2013.
- Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco, Luis Javier García Villalba, Julio Hernandez-Castro, Stuart James Gibson. Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform, Proceedings of the 5th International Conference on Crime Detection and Prevention (ICDP 2013), pages 1–6, London, UK, December 16 – 17, 2013.
- Ana Lucila Sandoval Orozco, David Manuel Arenas González, Luis Javier García Villalba, Julio Hernandez-Castro. Analysis of Errors in Exif Metadata on Mobile Devices. Multimedia Tools and Applications, 68(1): 1–29, January 2014.

- Ana Lucila Sandoval Orozco, David Manuel Arenas González, Jocelin Rosales Corripio, Luis Javier García Villalba, Julio César Hernández Castro: Source Identification for Mobile Devices, Based on Wavelet Transforms Combined with Sensor Imperfections. Computing, 96(9): 829–841, September 2014.

- David Manuel Arenas González, Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Hiram Jafet Romo Torres, Luis Javier García Villalba: Identificación de la Fuente en Videos de Dispositivos Móviles. Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pages 265–270, Alicante, España, Septiembre 2 – 5, 2014.

- David Manuel Arenas González, Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Jorge Alberto Zapata Guridi, Luis Javier García Villalba: Clasificación sin Supervisión de Imágenes de Dispositivos Móviles. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pages 271–276, Alicante, España, Septiembre 2 – 5, 2014.

- Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco, Luis Javier García Villalba: Identificación de la Fuente de Imágenes de Dispositivos Móviles Basada en el Ruido del Sensor. Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información (RECSI 2014), pages 277–280, Alicante, España, Septiembre 2 – 5, 2014.

- Ana Lucila Sandoval Orozco, Luis Javier García Villalba, David Manuel Arenas González, Jocelin Rosales Corripio, Julio César Hernández Castro and Stuart Gibson. Smartphone Image Acquisition Forensics Using Sensor Fingerprint. IET Computer Vision (en prensa), 2015.

Actas

XII Reunión Española sobre Criptología y Seguridad de la Información

Donostia-San Sebastian
2012

Editores:
U. Zurutuza
R. Uribeetxeberria
I. Arenaza-Nuño

4–7 Septiembre, 2012

Edita:
Servicio Editorial de Mondragon Unibertsitatea
http://recsi2012.mondragon.edu
Mondragon Unibertsitatea
Loramendi, 4. Apartado 23
20500 Arrasate - Mondragon

ISBN: 978-84-615-9933-2

1^a Edición: Julio de 2012

Anomalías en el Seguimiento de Exif en el Análisis Forense de Metadatos de Imágenes de Móviles

Ana Lucila Sandoval Orozco[1], David Manuel Arenas González[1], Luis Javier García Villalba[1], Julio César Hernández Castro[2]

[1] Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento. de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid
Email: {asandoval, darenas, javiergv}@fdi.ucm.es

[2] School of Computing, Buckingham Building, Lion Terrace, Portsmouth University
Portsmouth PO1 3HE, Reino Unido
Email: Julio.Hernandez-Castro@port.ac.uk

***Resumen*—Hoy en día el número de cámaras fotográficas integradas en dispositivos móviles crece a un ritmo imparable, haciendo necesario el uso de técnicas de análisis forense específicas para las imágenes generadas por este tipo de dispositivos, dada la singularidad de los mismos. La mayoría de estos dispositivos insertan metadatos Exif en el proceso de adquisición de la imagen y aun siendo estos fácilmente vulnerables a distintos tipos de modificaciones, son de gran ayuda para una gran variedad de técnicas de análisis forense. Teniendo todo esto en cuenta, se estima necesario la existencia de herramientas eficaces y robustas, que permitan la extracción de los metadatos de una forma veraz y consistente. Igualmente esta extracción de metadatos no debe manipular en ningún momento la imagen y requiere tener en cuenta posibles violaciones de la especificación Exif, tanto en la inserción de los metadatos en el proceso de adquisición de la imagen por parte de los fabricantes, como por parte de cualquier modificación, ya sea malintencionada o no. En este artículo se muestran anomalías en el seguimiento de la especificación Exif, lo que puede producir graves problemas en la extracción de los metadatos de las imágenes por medio de aplicaciones, así como problemas de interoperabilidad entre distintos dispositivos. Asimismo, se muestran anomalías en el funcionamiento de diversas herramientas forenses.**

I. Introducción

Actualmente, el número de cámaras integradas en dispositivos móviles supera a las cámaras digitales tradicionales (DSCs). Existen razones para ver que esta extensión de las cámaras fotográficas en dispositivos móviles son beneficiosas para las distintas situaciones en las que se requiere una prueba gráfica de un hecho (pruebas de delitos, privación de la libertad de prensa, etc.). Dadas las características técnicas particulares de este tipo de dispositivos, se necesitan herramientas de análisis forense específicas, no siendo válidas las herramientas que tratan imágenes de forma general o las generadas por otro tipo de dispositivos (DSCs, escáneres, etc.). Este trabajo está estructurado en 8 secciones, siendo la primera de ellas la presente introducción. La sección II realiza un estado del arte del análisis forense para imágenes generadas por dispositivos móviles haciendo un compendio de las principales técnicas utilizadas. En la sección III se realiza una descripción de los principales sistemas de metadatos en imágenes dando una especial importancia al estándar Exif que se detalla en la sección IV por su alto grado de utilización en imágenes generadas por dispositivos móviles. En la sección V se realiza un análisis binario de los metadatos de imágenes reales de varios teléfonos móviles. Este estudio permite una comprensión más a fondo del estándar Exif, así como examinar el cumplimiento de la especificación Exif por parte del los fabricantes. En la sección VI se describen algunos de los casos de violaciones encontrados en el seguimiento de la especificación Exif. En la sección VII se realiza un estudio comparativo de herramientas de análisis forense enfocadas al análisis de metadatos para evaluar la fiabilidad de la información Exif con inconsistencias en el seguimiento del estándar. Por último en la sección VIII se presentan las conclusiones y el trabajo futuro.

II. Técnicas de Análisis Forense de Imágenes

El área del análisis forense de imágenes puede dividirse en dos grandes ramas: autenticidad de las imágenes y la identificación de la fuente de creación de la imagen [1]. Con respecto a la primera de las ramas, nos referimos a determinar si una imagen no ha sufrido ningún procesamiento posterior al de su creación, es decir que no haya sido manipulada. La segunda de las ramas apunta a la identificación de la fuente de creación de la imagen. Las técnicas de esta rama se fundamentan en el estudio de las características del proceso de adquisición del dispositivo concreto y de la tecnología utilizada.

Aun teniendo en cuenta estas dos grandes ramas no se puede dejar pasar por alto la información de los metadatos que los dispositivos introducen en el proceso de adquisición de la fotografía. Suponiendo la veracidad de los datos contenidos en la imagen, es decir, que no se hayan dado manipulaciones mal intencionadas a posteriori, dependiendo de cada fabricante y dispositivo se arroja en una diversidad de formatos, una información útil para el analista forense (localización GPS, fuente de la foto, características técnicas de la imagen, etc.). Las

Análisis Forense de Imágenes de Dispositivos Móviles Utilizando los Metadatos Exif

Ana Lucila Sandoval Orozco[(1)], David Manuel Arenas González[(1)], Luis Javier García Villalba[(1)], Julio César Hernández Castro[(2)]
{asandoval, darenas, javiergv}@fdi.ucm.es, Julio.Hernandez-Castro@port.ac.uk
[(1)] Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento. de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid
[(2)] School of Computing, Buckingham Building, Lion Terrace, Portsmouth University
Portsmouth PO1 3HE, Reino Unido

***Resumen*—The quality and performance of cameras in mobile devices of common use are edging slowly to Digital Cameras. Thus, the forensic analysis of such images is particularly important and necessary in many situations (like, evidence in court cases, industrial espionage, deprivation of freedom of press, pederasty, etc.). Among the various branches of forensic analysis, this paper focuses on the acquisition of the source that produced the image. For this we have developed a technique based on Exif metadata, allowing in certain cases to obtain the power (make and model) with which the photo was taken. We have also developed a tool allowing various complex functions that help the forensic analyst, such as different types of advanced queries on Exif metadata information of large sets of images or functions of geopositioning.**

I. Introducción

Actualmente, incluso con el impacto de la crisis financiera global, el número de ventas de dispositivos móviles (teléfonos móviles, smartphones, PDAs, tablets, etc.) sigue aumentando. La inmensa mayoría, concretamente el 75 % de los dispositivos móviles vendidos en 2009, poseen una cámara fotográfica integrada [1]. Este tipo de cámaras integradas en dispositivos móviles ya superan en número a las cámaras digitales tradicionales (DSCs). El número de ventas de este tipo de cámaras fotográficas para 2011 supera los mil millones de unidades y se estima en unos mil trescientos millones para 2012 [2]. De igual modo, existen predicciones que indican que las DSCs desaparecerán en pro de las integradas en dispositivos móviles [3], ya que el aumento de la calidad de estas cámaras crece a un ritmo imparable.

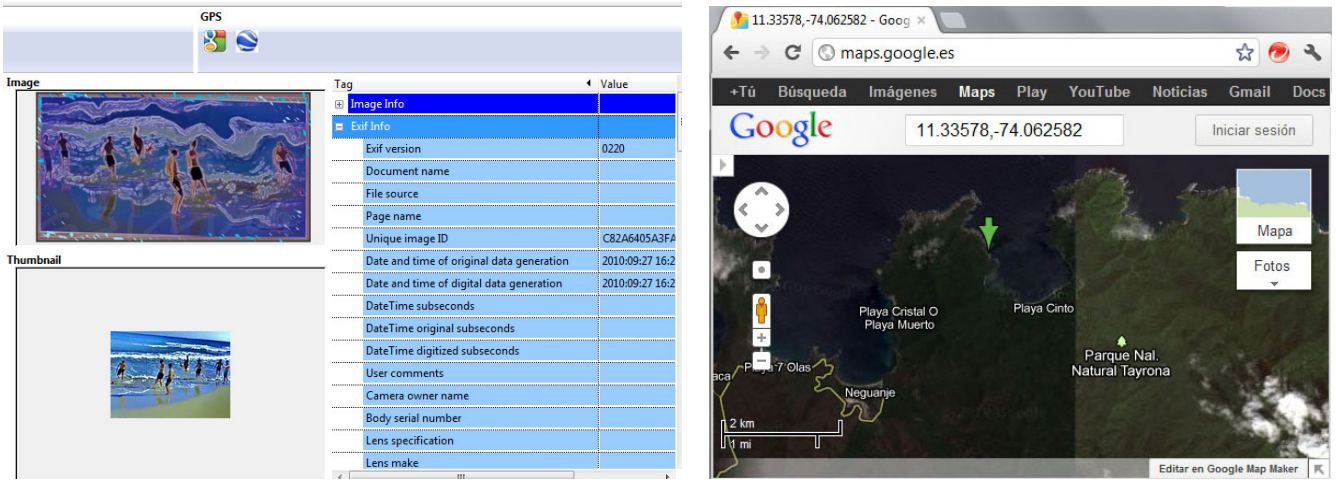
Asimismo, no sólo debe medirse en cifras de ventas la irrupción de las cámaras de dispositivos móviles en la sociedad actual. En nuestro día a día es habitual ver como se realizan y usan fotografías de este tipo de dispositivos para una gran diversidad de situaciones (vida personal, noticias, pruebas judiciales, aplicaciones para teléfonos móviles, etc.).

Muchos estiman que este tipo de cámaras facilitan la proliferación de crímenes contra la privacidad y la seguridad de la información (robo con tarjetas de crédito, pornografía infantil, espionaje industrial, etc.). De hecho, una de las principales razones de la existencia hoy en día de dispositivos móviles sin cámaras fotográficas se debe a que diversas compañías, organizaciones o gobiernos poseen normas que prohíben o limitan su uso [4]. Por otro lado existen razones para ver que esta inmensa extensión de las cámaras fotográficas en dispositivos móviles son beneficiosas para las distintas situaciones en las que se requiere una prueba gráfica de un hecho (pruebas de delitos, privación de la libertad de prensa, etc.). Por todo ello, es necesario proveer a los analistas forenses, de herramientas integrales que faciliten su tarea para todo tipo de investigaciones. Dadas las características técnicas particulares de este tipo de fotografías, estas herramientas de análisis forense deben ser específicas, no siendo válidas las herramientas que tratan imágenes de forma general o las generadas por otro tipo de dispositivos (DSCs, escáneres, etc.).

Este artículo está estructurado en 5 secciones, siendo la primera de ellas la presente introducción. La sección II comenta brevemente la técnicas de análisis forense de imágenes en dispositivos móviles a partir de los metadatos Exif por el alto grado de utilización de este estándar en las imágenes generadas por dispositivos móviles. La sección III presenta una herramienta para el análisis forense de imágenes de dispositivos móviles utilizando los metadatos Exif. En la sección IV se realiza una comparativa de esta herramienta con otras existentes. Por último, en la sección V se presentan las conclusiones y el trabajo futuro.

II. Metadatos Exif en Imágenes

El área del análisis forense de imágenes puede dividirse en dos grandes ramas: autenticidad de las imágenes y la identificación de la fuente de creación de la imagen [5]. Aun teniendo en cuenta estas dos grandes ramas no se puede dejar pasar por alto la información de los metadatos que los dispositivos introducen en el proceso de adquisición de la fotografía. Suponiendo la veracidad de los datos contenidos en la imagen, es decir, que no se hayan dado manipulaciones mal intencionadas a posteriori, dependiendo de cada fabricante y dispositivo se arroja en una diversidad de formatos, una información útil para el analista forense (localización GPS, fuente de la foto, características técnicas de la imagen, etc.). Las técnicas basadas en metadatos son las más sencillas, aunque dependen en gran medida de los datos que el fabricante decida insertar como metadatos en la imagen en el momento de la toma. Asimismo, este método es el más vulnerable a posibles cambios malintencionados por terceros. Aún así una vez que se pueda comprobar por distintos métodos o situaciones que no ha habido ningún tipo de manipulación externa, el análisis



(a) *Metadatos Exif.* (b) *Geoposicionamiento en Google Maps de una foto.*

Fig. 1. Análisis individual de fotos.

de la gran cantidad de metadatos que, actualmente como norma general insertan los fabricantes, puede ser de gran ayuda para las funciones del analista forense. Existe una gran variedad de trabajos que hacen referencia a los distintos tipos de metadatos en las imágenes con fines de búsquedas y clasificación [6] [7] [8]. Concretamente, para identificación de fuente de la cámara la especificación más seguida por la mayoría de los fabricantes, Exif, cuenta con dos etiquetas concretas "Make", para la marca y "Model" para el modelo. En ninguna de las versiones de la especificación Exif (hasta la versión 2.3) es obligatorio la existencia de estos dos campos. Asimismo si se incluyen pueden aparecer con valores vacíos indicando de esta forma que los valores son desconocidos.

III. Herramienta para el Análisis Forense de Imágenes de Dispositivos Móviles Utilizando los Metadatos Exif

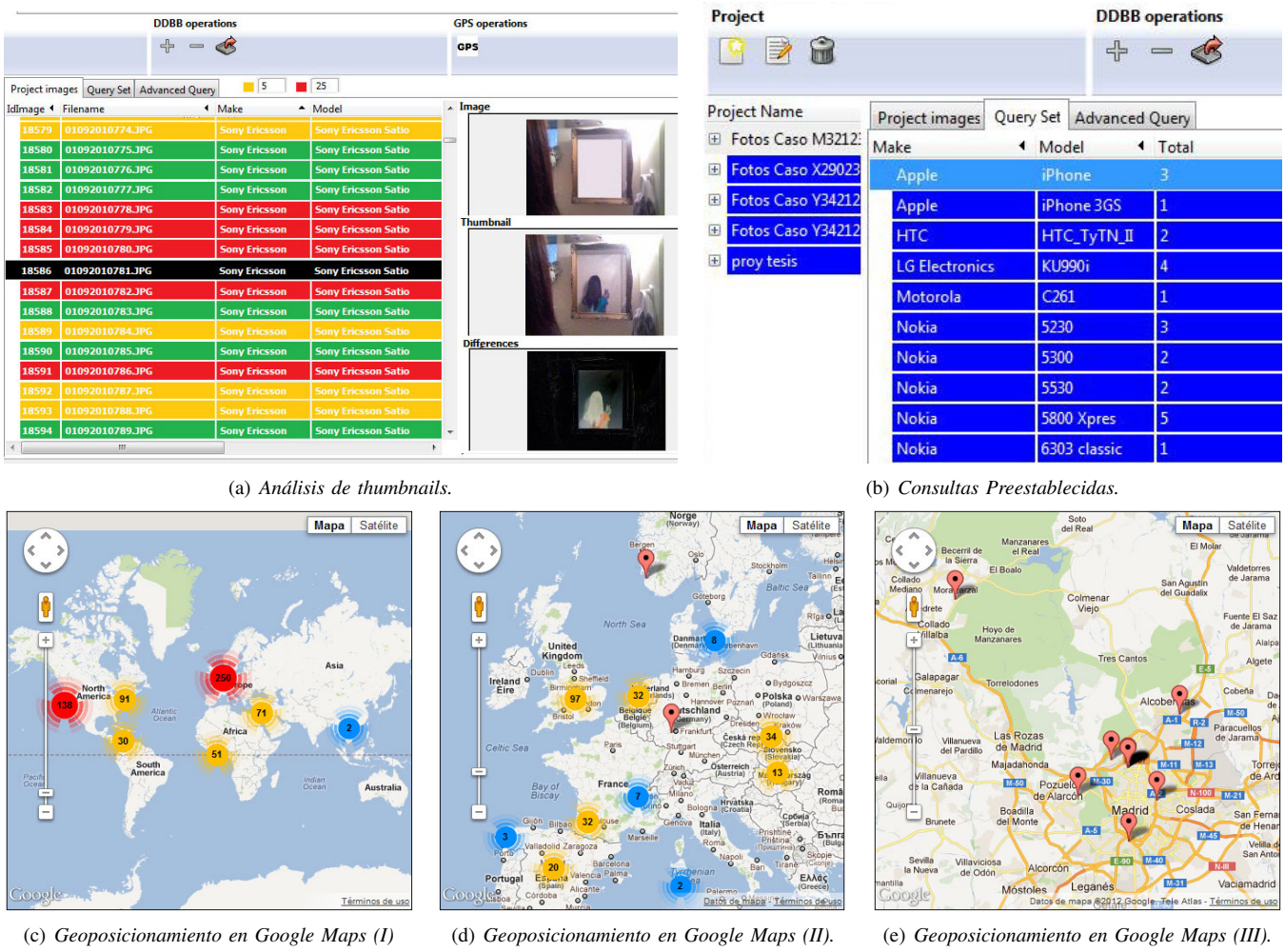
La herramienta desarrollada realiza dos tipos de análisis de imágenes: a nivel individual y a nivel grupal. La primera permite obtener la información Exif detallada de una imagen individual (determinación de la marca y modelo, entre otras), encontrar modificaciones realizadas a la foto al compararla con el *thumbnail* existente en la información Exif y situar la foto en Google Maps y Google Earth (si posee información de geoposicionamiento), como se puede observar en la Figuras 1(a) y 1(b). La segunda, permite hacer análisis de imágenes en su conjunto. Lo primero a destacar en esta funcionalidad es que las imágenes se analizan en grupos. Cada grupo es totalmente independiente entre sí. Los diferentes análisis que se pueden realizar sobre cada grupo son los siguientes: administración de imágenes (añadir y eliminar imágenes), consultas preestablecidas, análisis de modificaciones basadas en el *thumbnail* almacenado, consultas avanzadas y geoposicionamiento de las imágenes.

- **Análisis de modificaciones basado en el** *thumbnail*: Tiene como objetivo determinar si se realizaron modificaciones en las imágenes del grupo posteriores a la captura de las mismas. Esta operación se realiza calculando el root mean square de la comparación del *thumbnail* que se encuentra en la información Exif con el *thumbnail* generado a partir de imagen analizada: Clasificando las imágenes en: sin modificaciones, posiblemente modificadas y con modificaciones, representadas con los colores verde, naranja y rojo respectivamente como se muestra en la Figura 2(a).
- **Consultas preestablecidas**: Permite crear consultas agregando etiquetas Exif sobre las imágenes del grupo seleccionado. La consulta agrupa las imágenes por los criterios seleccionados y muestra el número de imágenes que hay en cada uno de los grupos formados como Figura 2(b).
- **Consultas avanzadas**: Permite la creación de consultas sobre imágenes de un grupo configurando los datos Exif mostrar y los filtros a aplicar. Es decir, muestra la información de las imágenes de los campos seleccionados que coincidan con uno de los valores de cada uno de los filtros configurados. Asimismo se permite el almacenamiento permanente de consultas.
- **Geoposicionamiento**: Permite mostrar la información de geoposicionamiento de un grupo de imágenes. Esta opción permite la selección de algunas o todas las imágenes con información de geoposicionamiento para la creación de un mapa en Google Maps que sitúe a las mismas. En el mapa se agrupan las fotos por zona, y a medida que se aumenta el zoom se va detallando las coordenadas. Las Figuras 2(c), 2(d) y 2(e) muestran un ejemplo del mapa generado y el proceso de aumento del zoom en una zona concreta.

IV. Comparativa con Otras Herramientas

Para realizar la comparativa de la herramienta desarrollada se han buscado herramientas de extracción y tratamiento de metadatos Exif para archivos JPEG, aunque no ha sido un criterio que excluya a otro tipo de herramientas relacionadas. Las herramientas son: *PhotoInfoEx*, *Exif Viewer*, *ExifPro Image Viewer*, *ExifTool* y *Jhead*.

PhotoInfoEx. Es un programa de fotografía digital que permite editar o modificar ciertos metadatos de la información Exif o IPTC de los archivos de imágenes en formato JPEG



(a) *Análisis de thumbnails.* (b) *Consultas Preestablecidas.*

(c) *Geoposicionamiento en Google Maps (I)* (d) *Geoposicionamiento en Google Maps (II).* (e) *Geoposicionamiento en Google Maps (III).*

Fig. 2. Análisis de un grupo de fotos.

y TIFF, así algunos archivos tipo de RAW. Las principales ventajas sobre la herramienta desarrollada son la mejor navegación sobre los archivos a examinar, la exportación de los metadatos obtenidos a Microsoft Excel y Microsoft Word y la impresión de los mismos. Como inconvenientes con respecto a la herramienta desarrollada se destaca los problemas en la extracción de metadatos Exif que no son acordes al 100 % con la especificación. Por ejemplo se han detectado imágenes con datos en la etiqueta "DateTimeOriginal" erróneos para la fecha. *PhotoInfoEx* en lugar de mostrar un error o la cadena tal como está almacenada, formatea los datos internamente y muestra otros distintos a los que posee la imagen, aparentemente correctos cuando realmente no lo son. Adicionalmente no permite ningún tipo de análisis grupal de fotos siendo éste un aspecto clave y por último no muestra información referente al *thumbnail* almacenado en la información Exif por lo que no permite determinar si hay modificaciones con respecto a la imagen original.

JHead. Es una herramienta de línea de comandos muy potente que permite extraer y manipular la información Exif de los archivos JPEG. La única ventaja destacable de *JHead* frente a la aplicación desarrollada es que permite la extracción de los metadatos IPTC y XMP (aunque estos no sean utilizados por los dispositivos móviles). El principal inconveniente es que, no permite el análisis grupal de fotos, lo cual es fundamental. Carece de interfaz gráfica lo cual dificulta su uso, no posee funciones de geoposicionamiento en Google Maps y Google Earth y no realiza análisis de *thumbnails*.

ExifTool. Es una herramienta que permite la extracción y edición de metadatos en una gran variedad de formatos de archivos. Soporta formatos de metadatos tales como Exif, IPTC, XMP, JFIF. Además permite decodificar información propia de los fabricantes "maker note" de gran cantidad de cámaras. Básicamente las ventajas e inconvenientes de ExifTool con respecto a la herramienta desarrollada son los mismos que con *JHead* y como consecuencia las conclusiones de la comparación.

Exif Viewer. Es un complemento para el navegador Firefox que permite extraer metadatos Exif, IPTC y XMP, de imágenes JPEG tanto locales como remotas. La principal ventaja de *Exif Viewer* con respecto a la aplicación desarrollada es su facilidad y rapidez en la instalación, así como la facilidad de uso (teniendo en cuenta las grandes limitaciones que tiene). Otra de las ventajas es permite el geoposicionamiento además de en Google Maps y Google Earth en Yahoo! Maps y en MSN Maps & Directions. El principal inconveniente, al igual que con todas las herramientas anteriormente comparadas, es que no permite un análisis de las imágenes en grupo, el cual insistimos clave. Asimismo la forma de presentar la información y el interfaz es austero y poco amigable. Por

tanto, se concluye que es una herramienta con muchas menos posibilidades que la herramienta desarrollada.

ExifPro Image Viewer. Permite mostrar la información Exif (un número muy limitado de tags) de imágenes JPEG. La principal ventaja de esta aplicación sobre todas las anteriormente tratadas (incluida la que se ha desarrollado) es el navegador de archivos de las imágenes. Ofrece una inmensa cantidad de posibilidades para mostrar, agrupar y ordenar las imágenes de los directorios. Asimismo es la más potente con respecto a la forma de mostrar las imágenes individuales. Con respecto a la extracción y tratamiento de los metadatos, sin duda, es la que más carencias tiene. Apenas extrae una veintena de tags Exif, los cuales presenta de una forma poco clara. No posee ningún tipo de funcionalidad de geoposicionamiento. Además de no permitir el tratamiento grupal de los metadatos de las imágenes y no analiza el *thumbnail* almacenado en la información Exif. Posee una opción que posibilita la exportación de la información Exif incluida en fotografías JPEG a un fichero TXT. Esto facilita la posterior importación de la información del archivo a otros formatos de bases de datos u hojas de cálculo. Posteriormente a esto se pueden realizar consultas grupales sobre los datos exportados, pero la aplicación no permite directamente este tipo de operación, además de requerir al analista forense de conocimientos informáticos avanzados para realizar este tipo de tratamiento. Por tanto las conclusiones que se obtienen es que esta herramienta más que tratar metadatos Exif, es una herramienta para la visualización y clasificación de imágenes. El conjunto de datos Exif que obtiene es excesivamente limitado y en ninguna circunstancia es una aplicación válida para la tarea de análisis forense.

Una vez comparada la herramienta desarrollada una a una con otras con propósitos comunes, se puede concluir que, no habiendo ninguna que ofrezca todas las mejores posibilidades, sin duda la herramienta presentada en este trabajo es la que ofrece una mayor funcionalidad, potencia y versatilidad a la tarea del analista forense. Ninguna de las herramientas comparadas posee un tratamiento de imágenes en grupo, así como una extracción de metadatos Exif más completa y organizada. Esta herramienta no tiene como objetivo primordial la visualización de galerías de imágenes, sino favorecer y automatizar en la medida de lo posible la tarea del análisis forense para imágenes de dispositivos móviles con respecto a los metadatos. Este objetivo se consigue con mayor éxito que con cualquiera de las herramientas presentadas en la comparación.

V. Conclusiones y Trabajo Futuro

Una vez desarrollada y utilizada la herramienta y haber realizado estudios sobre los metadatos Exif de un banco propio de imágenes se pueden obtener una serie de conclusiones enfocadas en dos planos: valoración de la herramienta y valoración de los estudios realizados. Sobre la valoración de la herramienta se han mostrado las diferencias con respecto a otras a la hora de la extracción de metadatos. Algunas otras aplicaciones permiten un rango mayor de formatos de metadatos. Sin embargo, la mayoría de las imágenes de dispositivos móviles están en formato JPEG/Exif, por tanto no hace que la herramienta tenga grandes desventajas con respecto a otras en ese aspecto. La principal ventaja que podemos observar es que la herramienta tiene como principal objetivo cubrir las necesidades de las tareas de un analista forense. No se ha encontrado ninguna aplicación que pueda competir en este aspecto con la presentada, ya que ninguna permite realizar las siguientes operaciones:

- Análisis en grupos de imágenes, las herramientas estudiadas únicamente ofrecen información de metadatos de imágenes individuales.
- Comparar el *thumbnail* almacenado en los metadatos Exif de cada imagen para encontrar modificaciones posteriores a la captura de las fotos.
- Visualización en Google Maps de las imágenes en su conjunto.

Por un lado conviene recordar que todos los análisis realizados sobre metadatos Exif son desgraciadamente fácilmente vulnerables a modificaciones malintencionadas por terceros. No obstante, cabe destacar que este hecho no implica la inutilidad del uso de metadatos en imágenes, ya que actualmente se puede asegurar que son imprescindibles e inseparables en una imagen digital. Por otro lado, los metadatos aportan información útil para el analista forense como por ejemplo la relacionada con el geoposicionamiento, la cual actualmente es imposible inferir mediante el contenido de la imagen y la evidencia de modificaciones posterior a la captura de una foto, basados en el *thumbnail* almacenado en su información Exif. Asimismo la identificación de la fuente de la imagen con los metadatos depende totalmente de su inserción por parte del fabricante. Por tanto, es necesario añadir nuevas técnicas para la identificación de fuente de una imagen, a la vez que también son imprescindibles las funcionalidades presentadas que tratan los metadatos Exif.

Agradecimientos

Los autores agradecen la financiación que les brinda el Subprograma AVANZA COMPETITIVIDAD I+D+I del Ministerio de Industria, Turismo y Comercio (MITyC) a través del Proyecto TSI-020100-2011-165. Asimismo, los autores agradecen la financiación que les brinda el Programa de Cooperación Interuniversitaria de la Agencia Española de Cooperación Internacional para el Desarrollo (AECID), Programa PCI-AECID, a través de la Acción Integrada MAEC-AECID MEDITERRÁNEO A1/037528/11.

Referencias

[1] J. Hsu, "The Worldwide Mobile Phone Camera Module Market and Taiwan's Industry, 2009 and Beyond", en *Market Intelligence & Consulting Institute (MIC)*, 2009.

[2] Infotrends, "Worldwide Camera Phone Forecast: 2007-2012", en *Weymouth, MA: Infotrends*, 2008.

[3] Richard L. Baer, "Resolution Limits in Digital Photography: the Looming End of the Pixel Wars", en *Imaging Systems, OSA technical Digest (CD) (Optical Society of America*, 2010.

[4] L. Srivastava, "Mobile phones and the evolution of social behavior", en *Behaviour & Information Technology*, Vol. 24, No. 2, pp. 111–129, 2005.

[5] T. Gloe, M. Kirchner, A. Winkler, R. Böhme, "Can We Trust Digital Image Forensics?", en *Proceedings of the 15th International Conference on Multimedia (MM'07)*, Augsburg, Bavaria, Germany, September 23-28, pp. 78–86. ACM Press, New York, 2007.

[6] M. Boutell, J. Luo, "Photo classification by integrating image content and camera metadata", en *Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04)*, Vol. 4, pp. 901–904, 2004.

[7] J. Tesic, "Metadata Practices for Consumer Photos", en *IEEE Multimedia*, Vol. 12, No. 3, pp. 86–92, 2005.

[8] M. Boutell, J. Luo, "Beyond: pixels: Exploiting camera metadata for photo classification", en *Pattern Recognition*, Vol. 38 No. 6, pp. 935–946, 2005.

TECHNIQUES FOR SOURCE CAMERA IDENTIFICATION

Ana Lucila Sandoval Orozco[1], Jocelin Rosales Corripio[1], David Manuel Arenas González[1], Luis Javier García Villalba[1] Julio César Hernández Castro[2]

[1] Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
Email: {asandoval, darenas, javiergv}@fdi.ucm.es, jocelinr@estumail.ucm.es

[2] School of Computing, Office S129A, University of Kent
Cornwallis South Building, Canterbury CT2 7NF, UK
E-mail J.C.Hernandez-Castro@kent.ac.uk

Abstract

Digital image forensics has lately become one of the very important applications to identify the characteristics and the originality of the digital devices. This paper studies the recent developments in the field of image source identification. Proposed techniques in the literature are categorized into five primary areas based on source model identification: Metadata, Image Features, CFA and Demosaicing Artifacts, Lens Distortions and Wavelet Transforms. The main idea of the proposed approaches in each category is described in detail, and reported results are discussed to evaluate the potential of the methods.

Keywords - Image forensics, source camera identification, classification, SVM.

1 INTRODUCTION

Image source identification research investigates the design of techniques to identify the characteristics of digital data acquisition device (e.g., digital camera and cell-phone) used in the generation of an image. These techniques are expected to achieve two major outcomes. The first is the class (model) properties of the source, and the second is the individual source properties.

The success of image source identification techniques depends on the assumption that all images acquired by an image acquisition device will exhibit certain characteristics that are intrinsic to the acquisition devices because of their (proprietary) image formation pipeline and the unique hardware components they deploy, regardless of the content of the image. (It should be noted that such devices generally encode the device related information, like model, type, date and time, and compression details, in the image header, e.g., EXIF header. However, since this information can be easily modified or removed, it cannot be used for forensics purposes).

1.1 Image Formation in Digital Cameras

The design of image source identification techniques requires an understanding of the physics and operation of these devices. The general structure and sequence of stages of image formation pipeline remains similar for almost all digital cameras, although much of the details are kept as proprietary information of each manufacturer.

Consumer level digital cameras consist of a lens system, sampling filters, colour filter array, imaging sensor and a digital image processor [1]. The lens system is essentially composed of a lens and the mechanisms to control exposure, focusing, and image stabilization to collect and control the light from the scene. After the light enters the camera through the lens, it goes through a combination of filters that includes at least the infra-red and anti-aliasing filters to ensure maximum visible quality. The light is then focused onto imaging sensor, an array of rows of columns of light-sensing elements called pixels. Digital cameras deploy charge-coupled device (CCD) or complimentary metal-oxide

Source Smartphone Identification Using Sensor Pattern Noise and Wavelet Transform

Jocelin Rosales Corripio[1], David Manuel Arenas González[1], Ana Lucila Sandoval Orozco[1], Luis Javier García Villalba[1], Julio Hernandez-Castro[2], Stuart James Gibson[3]

[1] Group of Analysis, Security and Systems (GASS)
Department of Software Engineering and Artificial Intelligence (DISIA)
School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
[2] School of Computing, University of Kent, Canterbury CT2 7NF, UK
[3] School of Physical Sciences, University of Kent
Canterbury, Kent, United Kingdom, CT2 7NH

Keywords: Digital Image, Forensics Analysis, Photo Response Non Uniformity, PRNU.

Abstract

The ability to relate a digital photograph to its source camera has application in the areas of digital forensics and multimedia data mining. The majority of previous research in this area has focused on primary function imaging devices (i.e. digital cameras). In this work we use the pattern noise of an imaging sensor to classify digital photographs according to the source smartphone from which they originated. This is timely work as new smartphone models large imaging sensors, afford significant improvements in classification rates using pattern noise. Our approach is to extract wavelet based features which are then classified using a support vector machine. We show that this method generalises well when the number of source cameras is increased.

1 Introduction

Due to increasing storage capacity, usability, portability and affordability, camera enabled mobile phones have become ubiquitous consumer electronic devices. The extensive use of smartphone cameras makes enforcing legal restrictions on the capture and sharing of digital photographs very difficult. Restrictions on the use of cameras include locations such as schools, government offices and businesses. Consequently, tools which permit the identification of source devices have significant utility various areas of law enforcement [2] such as child protection and digital rights management.

2 Source Camera Identification Techniques

Research in this field typically determines make and model by identifying characteristic artefacts within an image. The success of these techniques depend on the assumption that the characteristics are unique to each device [21]. The main problem with this approach is that different models of digital camera are often built using the same core components that originate from a small number of manufacturers. As a consequence it can be difficult, or in some cases impossible, to differentiate between models using such methods.

Numerous approaches to the camera identification have been explored. It has been suggested [8] that the lens radial distortion is the best technique for source identification. Radial distortion causes straight lines to appear as curves in images. The degree of radial distortion for each image can be measured by a process consisting of three steps: edge detection, distorted segment extraction, and distortion error measurement. They experimented with three different cameras and obtained 91.28% source camera identification accuracy.

In [3] an algorithm for identifying and classifying color interpolation operations is presented. The method comprises two algorithms: the first algorithm analyses the correlation of each pixel value with values of its neighbouring pixels, the second analyses the differences between pixels independently. The source camera identification results with images from four to five different models resulted an accuracy of 88% and 84.8% respectively.

Between pixel correlations for source identification were also used in [14], obtaining a coefficient matrix for each color channel while defining a pixel quadratic correlation model. A neural network classifier was used, achieving a success rate of 98.6%. This method is not effective in differentiating between models originating from the same manufacturer.

A set of binary similarity measures is used in [4] as metrics to estimate the similarity between image bit planes. The fundamental assumption of this work is that colour filter array *Color Filter Array* (CFA) demosaicing algorithms, from each make, leave correlations along image bit planes and can be represented by a set of 108 binary similarity measures for classification. The success rate of the experiment was between 81% and 98% when attempting to classify three cameras which

Multimed Tools Appl
DOI 10.1007/s11042-013-1837-6

Analysis of errors in exif metadata on mobile devices

Ana Lucila Sandoval Orozco · David Manuel Arenas González · Luis Javier García Villalba · Julio Hernández-Castro

Abstract Nowadays the number of cameras integrated in mobile phones is growing very fast, making it essential to design new specific forensic analysis techniques aimed towards the pictures created with these devices. Most of these phones automatically add relevant Exif metadata in the process of image acquisition. This metadata, even if it is vulnerable to tampering, can be very helpful for a variety of forensic analysis techniques. That is why the existence of efficient, robust and specialized tools is a necessity. These should allow metadata to be extracted in a consistent, fast and sound way. Besides, metadata extraction must never manipulate the image and it needs to take into account possible departures from the Exif specification, including the insertion of the metadata in the image acquisition process by the makers, as well as any modification, whether malicious or not. This paper will show the multiple anomalies in the Exif specification we have found during our study, which can produce serious problems in classical tools for the extraction of image metadata, including crashes and wrong results, and even interoperability problems among different devices. We will also show some anomalies found in the operation of different well-known forensic tools.

Keywords Camera mobile phones · Exif metadata · Forensics analysis

A. L. Sandoval Orozco · D. M. Arenas González · L. J. García Villalba (✉)
Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
e-mail: javiergv@fdi.ucm.es

A. L. Sandoval Orozco
e-mail: asandoval@fdi.ucm.es

D. M. Arenas González
e-mail: darenas@fdi.ucm.es

J. Hernández-Castro
School of Computing, Office S129A, University of Kent, Cornwallis South Building, Canterbury CT2 7NF, UK
e-mail: J.C.Hernandez-Castro@kent.ac.uk

Published online: 11 January 2014

Computing (2014) 96:829–841
DOI 10.1007/s00607-013-0313-5

Source identification for mobile devices, based on wavelet transforms combined with sensor imperfections

A. L. Sandoval Orozco · D. M. Arenas González · J. Rosales Corripio · L. J. García Villalba · J. C. Hernandez-Castro

Received: 22 January 2013 / Accepted: 11 February 2013 / Published online: 28 February 2013

Abstract One of the most relevant applications of digital image forensics is to accurately identify the device used for taking a given set of images, a problem called source identification. This paper studies recent developments in the field and proposes the mixture of two techniques (Sensor Imperfections and Wavelet Transforms) to get better source identification of images generated with mobile devices. Our results show that Sensor Imperfections and Wavelet Transforms can jointly serve as good forensic features to help trace the source camera of images produced by mobile phones. Furthermore, the model proposed here can also determine with high precision both the brand and model of the device.

Keywords Image forensics · Source model identification · Classification · Wavelet · Sensor imperfection · Support vector machines (SVMs)

A. L. Sandoval Orozco · D. M. Arenas González · J. Rosales Corripio · L. J. García Villalba (✉)
Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
e-mail: javiergv@fdi.ucm.es

A. L. Sandoval Orozco
e-mail: asandoval@fdi.ucm.es

D. M. Arenas González
e-mail: darenas@fdi.ucm.es

J. Rosales Corripio
e-mail: jocelinr@estumail.ucm.es

J. C. Hernandez-Castro
School of Computing, Office S129A, University of Kent, Cornwallis South Building, Canterbury CT2 7NF, UK
e-mail: J.C.Hernandez-Castro@kent.ac.uk

Rafael Álvarez · Joan Josep Climent · Francisco Ferrández · Francisco M. Martínez
Leandro Tortosa · José Francisco Vicent · Antonio Zamora
(editores)

Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información

RECSI XIII

Alicante, 2-5 de septiembre de 2014

PUBLICACIONES UNIVERSIDAD DE ALICANTE

Actas de la XIII Reunión Española sobre Criptología y Seguridad de la Información

RECSI XIII

Alicante, 2-5 de septiembre de 2014

Rafael Álvarez · Joan Josep Climent · Francisco Ferrández · Francisco M. Martínez
Leandro Tortosa · José Francisco Vicent · Antonio Zamora
(editores)

Publicaciones de la Universidad de Alicante

Campus de San Vicente, s/n
03690 San Vicente del Raspeig
Publicaciones@ua.es - http://publicaciones.ua.es
Teléfono: 965 903 480

ISBN: 978-84-9717-323-0

Universitat d'Alacant
Universidad de Alicante

Identificación de la Fuente en Vídeos de Dispositivos Móviles

David Manuel Arenas González, Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco, Hiram Jafet Romo Torres, Luis Javier García Villalba

Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {darenas, jocerosa, asandoval, javiergv}@fdi.ucm.es, hromot@hotmail.com

***Resumen*—La realización de vídeos con dispositivos móviles se ha convertido en una actividad común dado su alto grado de utilización y el gran número de usuarios. Además, la portabilidad de este tipo de dispositivos hace que estén a mano de los usuarios gran cantidad de tiempo facilitando que se utilicen para generar vídeos en una gran diversidad de situaciones. Por tanto, estos vídeos pueden ser utilizados como evidencias en procesos judiciales. Todo lo anterior hace necesario contar con técnicas de análisis forense enfocadas en vídeos de dispositivos móviles dada las características peculiares de sus cámaras. En este trabajo se estudia la identificación de la fuente de adquisición de los vídeos de dispositivos móviles y se presenta una técnica basada en la extracción del ruido del sensor y la transformada wavelet de los fotogramas extraídos del vídeo. Estos fotogramas son extraídos mediante un algoritmo que tiene en cuenta la naturaleza de los mismos, mejorando la selección de los fotogramas a analizar. Finalmente se presentan experimentos con vídeos de dispositivos móviles para evaluar la validez de las técnicas utilizadas.**

Palabras clave*—Análisis forense de vídeos, fuente de adquisición de vídeos, patrón de ruido del sensor, PRNU. *(Video forensics analysis, video source adquisition, sensor pattern noise, PRNU).

I. Introducción

Si las imágenes capturadas por dispositivos electrónicos son consideradas parte de la verdad como hechos reales, en pocos minutos, un vídeo puede comunicar una enorme cantidad de información. Según el medidor de tráfico "*Alexa, The Web Information Company*" [1], Youtube es actualmente el tercer sitio con más visitas del mundo, lo cual nos deja un claro indicio de la popularidad de la que gozan los vídeos entre los diferentes medios en los que puede desplegarse. Existe una amplia gama de dispositivos móviles que pueden reproducirlo y/o grabarlo, como por ejemplo: teléfonos móviles, tablets, vídeoconsolas portátiles y cámaras digitales o de vídeo. En cuanto a los dispositivos móviles, *Gartner Inc.* [2], afirma que las ventas de teléfonos inteligentes creció un 36 % en el cuarto trimestre del 2013. Asimismo, este tipo de dispositivos representó el 57.6 % de las ventas globales de teléfonos móviles en el cuarto trimestre de 2013, frente al 44 % del año anterior. Al igual que las cámaras digitales han desplazado en términos de uso a las cámaras tradicionales de película, actualmente, los dispositivos móviles equipados con cámaras, tienen un papel importante poniendo fin al rápido crecimiento de las cámaras digitales. En los dispositivos móviles, se ha visto una gran competencia entre fabricantes que se esfuerzan en integrar una videocámara de alta definición al alcance del usuario. Como consecuencia de este fenómeno y de la gran cantidad de tiempo que una persona pasa junto a un teléfono inteligente, este se ha convertido en el primer dispositivo de grabación de vídeos para muchos usuarios en la sociedad actual.

Debido al frecuente uso de los dispositivos móviles, en ciertos casos existen restricciones legales sobre el uso de este dispositivo, así como también de su uso en distintos lugares, tales como: colegios, universidades, oficinas de gobierno, empresas, etc. Actualmente los vídeos se exhiben con mayor frecuencia, ya sea directa o indirectamente en procesos judiciales como pruebas o evidencias para la aplicación de la ley [3]. Por tanto, dada la importancia de los vídeos en estas situaciones, el análisis forense cobra especial relevancia. Dentro de las distintas ramas del análisis forense, destaca la que nos permite identificar la fuente de adquisición, en este caso de la videocámara que generó el vídeo. En este trabajo se presentan técnicas de análisis forense para la identificación de la fuente de adquisición de vídeos, centrándonos especialmente en los vídeos generados por dispositivos móviles.

Este trabajo está estructurado en 6 secciones, siendo la primera de ellas la presente introducción. En la sección 2 se presentan brevemente las diferencias entre el pipeline en la creación de una imagen y un vídeo. La sección 3 realiza un estado del arte del análisis forense para imágenes y vídeos generados por dispositivos móviles. En la sección 4 se presenta la técnica propuesta. Los experimentos realizados y sus resultados son presentados en la sección 5. Por último en la sección 6 se presentan las conclusiones obtenidas de este trabajo.

II. Pipeline de una Videocámara

Antes de mencionar alguna de las técnicas existentes para la identificación de la fuente, es importante comprender cuál es el procedimiento realizado para generar un vídeo. Este proceso es similar en la generación de una imagen y de un vídeo, salvo que en un vídeo finalmente existe un último paso que consiste

Clasificación sin Supervisión de Imágenes de Dispositivos Móviles

David Manuel Arenas González, Jocelin Rosales Corripio, Ana Lucila Sandoval Orozco,
Jorge Alberto Zapata Guridi, Luis Javier García Villalba

Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: {darenas, jocerosa, asandoval, javiergv}@fdi.ucm.es, jorge.zapata@jazg.net

***Resumen*—Cada día el uso de imágenes de dispositivos móviles como evidencias en procesos judiciales es más habitual y común. Por ello, el análisis forense de imágenes de dispositivos móviles cobra especial importancia. En este trabajo se estudia la rama del análisis forense que se basa en la identificación de la fuente, concretamente en la agrupación o *clustering* de imágenes según la fuente de adquisición. Como diferencia con otras técnicas del estado del arte para la identificación de la fuente, en el *clustering* no se tiene un conocimiento a priori del número de imágenes ni dispositivos a identificar, ni se tienen datos de entrenamiento para una futura fase de clasificación. Es decir, se realiza un agrupamiento por clases con todas las imágenes de entrada. La propuesta se basa en la combinación de *clustering* jerárquico y plano y en el uso del patrón de ruido del sensor. Se han realizado un conjunto de experimentos que emulan situaciones similares a las que se pueden dar en la realidad para mostrar la robustez y fiabilidad de los resultados de la técnica. Los resultados obtenidos son satisfactorios en todos los experimentos realizados superando en tasa de acierto a otras propuestas descritas en el estado del arte.**

***Palabras clave*—Análisis forense de imágenes, clustering de imágenes, patrón de ruido del sensor, PRNU. *(Image forensics analysis, image clustering, sensor pattern noise, PRNU)*.**

I. Introducción

En la actualidad, el número de cámaras integradas a dispositivos móviles ha proliferado permitiendo a millones de consumidores tomar fotografías e incluso compartir de manera sencilla el contenido capturado. La industria de los dispositivos móviles ha desarrollado la tecnología necesaria para abaratar los costos y de esta manera hacerlos muy accesibles al público.

El gran número de cámaras en dispositivos móviles constituye un mayor número de evidencias presentadas ante la ley en delitos como robo de información de tarjetas de crédito, pornografía infantil, espionaje industrial, etc. Por tanto, el análisis forense de este tipo de imágenes cobra especial importancia en las investigaciones judiciales. Dentro de análisis forense de imágenes digitales existen dos grandes ramas: la identificación de la fuente de adquisición y la detección de manipulaciones malintencionadas. Este trabajo se centra en la primera rama, es decir, dada una imagen o conjunto de imágenes identificar la marca y modelo de la cámara que realizó la foto mediante la clasificación por agrupamiento o *clustering*. Asimismo, dado que las cámaras de dispositivos móviles tienen unas características propias que las hacen diferentes a la restantes, este trabajo se enfoca en las fotos de este tipo de dispositivos.

Dentro de la identificación de la fuente existen dos grandes enfoques: escenarios cerrados o escenarios abiertos. Un escenario cerrado es aquel en el cual la identificación de la fuente de la imagen se realiza sobre un conjunto de cámaras concreto y conocidas a priori. Para este enfoque normalmente se utiliza un conjunto de imágenes de cada cámara para entrenar un clasificador y posteriormente se predice la fuente de adquisición de las imágenes objeto de investigación. La técnica más utilizada para la tarea de clasificación de imágenes digitales es *Support Vector Machine* (SVM). Este trabajo se centra en la identificación de la fuente en escenarios abiertos, es decir, el analista forense no conoce a priori el conjunto de cámaras a las que pertenece la imagen a identificar su fuente. Obviamente en este tipo de clasificación, en la que no se tienen datos de cámaras a priori, el objetivo no es identificar la marca y modelo de la cámara, sino poder agrupar distintas imágenes en grupos disjuntos en los que todas sus imágenes pertenecen al mismo dispositivo. Este planteamiento es muy cercano a situaciones de la vida real, ya que en muchos casos el analista desconoce por completo el conjunto de cámaras a las que pueden pertenecer un conjunto de imágenes. Además, es prácticamente imposible tener un conjunto de imágenes para entrenar un clasificador con todas las cámaras de dispositivos móviles existentes en el mundo.

Este trabajo está estructurado en 5 secciones, siendo la primera de ellas la presente introducción. En la sección 2 se presentan brevemente los trabajos previos relacionados con las técnicas de análisis forense para la identificación de la fuente de imágenes de dispositivos móviles. En la sección 3 se presenta la técnica propuesta. Los experimentos realizados y sus resultados se presentan en la sección 4. Por último en la sección 5 se presentan las conclusiones obtenidas de este trabajo.

II. Trabajos Relacionados

La mayoría de las investigaciones realizadas sobre la identificación de la fuente de adquisición de imágenes se centran en cámaras digitales tradicionales o DSC (*Digital Still Camera*), no siendo en su mayoría estas técnicas válidas para imágenes

Identificación de la Fuente de Imágenes de Dispositivos Móviles Basada en el Ruido del Sensor

Jocelin Rosales Corripio, David Manuel Arenas González, Ana Lucila Sandoval Orozco,
Luis Javier García Villalba

Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases, 9, Ciudad Universitaria, 28040 Madrid
Email: jocerosa@ucm.es, {darenas, asandoval, javiergv}@fdi.ucm.es

***Resumen*—La fuente de una imagen digital se puede identificar a través de los rasgos que el dispositivo que la genera impregna en ella durante el proceso de su generación. La mayoría de las investigaciones realizadas en los últimos años sobre técnicas de identificación de fuente se han enfocado únicamente en la identificación de cámaras tradicionales DSC (*Digital Still Camera*). Considerando que hoy en día las cámaras de los dispositivos móviles prácticamente han sustituido a las DSCs se detectó la necesidad de realizar investigación sobre las técnicas para identificar la fuente de imágenes generadas por dispositivos móviles. Las imágenes digitales generadas por una cámara digital contienen intrínsecamente un patrón del ruido del sensor que se puede usar como medio de identificación de la fuente. Específicamente, las cámaras digitales de dispositivos móviles cuentan en su mayoría con un tipo de sensor que deja rasgos característicos en la imagen. En este trabajo se propone un algoritmo basado en el ruido del sensor y en la transformada wavelet para identificar el dispositivo móvil (marca y modelo) que ha generado determinadas imágenes bajo investigación.**

Palabras clave*—Análisis forense, imagen digital, patrón de ruido del sensor, PRNU. *(Forensics analysis, digital image, sensor pattern noise, PRNU).

I. Introduction

Con frecuencia las fotografías son consideradas como una parte de la verdad al ser hechos reales capturados por dispositivos electrónicos (cámaras). Sin embargo, con el desarrollo de la tecnología han surgido herramientas potentes y sofisticadas que facilitan de una manera impresionante la alteración de las imágenes digitales, incluso para quienes no tienen conocimientos técnicos o especializados en el área [1].

El desarrollo de las tecnologías digitales ha estado y continúa avanzando a un ritmo imparable. Cada día el número de cámaras digitales va creciendo, así como la facilidad de acceso a ellas. Las cámaras digitales de móviles merecen especial atención, ya que estudios realizados indican que al final del año 2012 el número total de dispositivos móviles activos alcanzó los 6,7 billones y se estima que para el verano del 2013 este número igualará al total de la población del planeta 7,1 billones. El 83 % de estos dispositivos móviles cuentan con cámara digital integrada, las cuales a diferencia de las cámaras digitales convencionales son llevadas por sus dueños todo el tiempo a la mayoría de lugares que asiste y en muchos casos tienen conexión a internet [2].

Debido al incremento en sus capacidades de almacenamiento, procesamiento, usabilidad y portabilidad así como a su bajo coste, los dispositivos móviles están presentes en diversidad de actividades, lugares y eventos de la vida diaria. A causa del extenso uso de las cámaras digitales de dispositivos móviles se han generado polémicas, discusiones y normas sobre la prohibición de su uso en lugares como escuelas, oficinas de gobierno, eventos empresariales, conciertos, empresas, etc. Una consecuencia más de su extenso uso es que las imágenes digitales en la actualidad son utilizadas como testigos silenciosos en procesos judiciales, siendo una pieza crucial de la evidencia del crimen [3]. Es por ello que contar con herramientas que permitan identificar a los dispositivos que han generado una cierta imagen digital cobra importancia ya que podría servir en diversas áreas como la lucha contra la pornografía infantil, la prevención de robo de tarjetas de crédito, el combate a la piratería, la prevención de secuestros, etc.

II. Técnicas de Análisis Forense en Imagenes

La investigación en este campo estudia el diseño de técnicas para identificar las características, especialmente marca y modelo, de los dispositivos utilizados para la generación de imágenes digitales. El éxito de estas técnicas depende del supuesto de que todas las imágenes adquiridas por un mismo dispositivo presentan características intrínsecas del dispositivo. Las características que se usan para identificar marca y modelo de las cámaras digitales se derivan de las diferencias que existen entre las técnicas de procesamiento de las imágenes y las tecnologías de los componentes que se utilizan [4]. El mayor problema con este enfoque es que los diferentes modelos de las cámaras digitales usan componentes de un número reducido de fabricantes, y que los algoritmos que usan también son muy similares entre modelos de la misma marca. Es por ello que la fiabilidad de la identificación de la cámara fuente depende en gran parte de la identificación de varias características independientes del modelo. Según [4] se pueden establecer cuatro grupos de técnicas para este fin: utilización de la aberración de las lentes, interpolación de la matriz CFA, uso de las características de la imagen e imperfecciones del sensor. Esta última constituye el objeto de

Smartphone Image Acquisition Forensics Using Sensor Fingerprint

Ana Lucila Sandoval Orozco[a], Luis Javier García Villalba[a,], David Manuel Arenas González[a], Jocelin Rosales Corripio[a], Julio Hernandez-Castro[b], Stuart James Gibson[c]

[a]*Group of Analysis, Security and Systems (GASS)*
Department of Software Engineering and Artificial Intelligence (DISIA)
School of Computer Science, Office 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid, Spain
[b]*School of Computing, University of Kent, Canterbury CT2 7NF, UK*
[c]*School of Physical Sciences, University of Kent*
Canterbury, Kent, United Kingdom, CT2 7NH

Abstract

The forensic analysis of digital images from mobile devices is particularly important given the quick expansion and everyday use of them in our society. A further consequence of digital images widespread use is that they are used today as silent witnesses in legal proceedings, as a crucial piece of evidence of the crime. This paper specifically addresses the description of a technique that allows the identification of the image source acquisition, for the specific case of mobile devices images. Our approach is to extract wavelet based features from sensor pattern noise which are then classified using a support vector machine. Moreover there are a number of parameters that allow us to adapt the execution of the algorithm to specific situations desired for the forensic analyst (a variety of types and sizes of image or optimizing the average accuracy rate in terms of processing time). This article describes a set of experiments with the same set of images that can obtain general conclusions for the different configurations.

Keywords: Digital Image, Forensics Analysis, Image Source Acquisition, Photo

Email addresses: asandoval@fdi.ucm.es (Ana Lucila Sandoval Orozco), javiergv@fdi.ucm.es (Luis Javier García Villalba), darenas@fdi.ucm.es (David Manuel Arenas González), jocerosa@ucm.es (Jocelin Rosales Corripio), J.C.Hernandez-Castro@kent.ac.uk (Julio Hernandez-Castro), s.j.gibson@kent.ac.uk (Stuart James Gibson)